



**TUGAS AKHIR - TE 145561**

**PEMBUATAN MODUL KOMUNIKASI PADA *MULTI-DISPLAY* YANG DIKONTROL SECARA TERPUSAT MENGGUNAKAN *WIRELESS***

Moch Nafi Rizky Kurniawan  
NRP. 2214030023

Dosen Pembimbing  
Eko Pramunanto, ST., MT.

PROGRAM STUDI KOMPUTER KONTROL  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017





**FINAL PROJECT - TE 145561**

**CREATING OF A COMMUNICATION MODULE ON  
*MULTI-DISPLAY* WHICH IS CONTROLLED  
CENTRALLY USING *WIRELESS***

Moch Nafi Rizky Kurniawan  
NRP. 2214030023

Advisor  
Eko Pramunanto, ST., MT.

COMPUTER CONTROL STUDY PROGRAM  
Electrical and Automation Engineering Department  
Vocational Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----

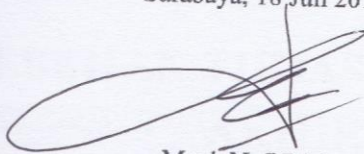
## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **"PEMBUATAN MODUL KOMUNIKASI PADA *MULTI-DISPLAY* YANG DIKONTROL SECARA TERPUSAT MENGGUNAKAN *WIRELESS*"** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 18 Juli 2017



Moch Nafi Rizky

NRP. 2214030023

-----Halaman ini sengaja dikosongkan-----

**PEMBUATAN MODUL KOMUNIKASI PADA MULTI-DISPLAY  
YANG DIKONTROL SECARA TERPUSAT MENGGUNAKAN  
WIRELESS**

**TUGAS AKHIR**

Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada  
Program Studi Komputer Kontrol  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing



Eko Pramunanto, ST., MT.  
NIP. 196612031994121001

**SURABAYA  
JULI, 2017**

-----Halaman ini sengaja dikosongkan-----



# **PEMBUATAN MODUL KOMUNIKASI PADA *MULTI-DISPLAY* YANG DIKONTROL SECARA TERPUSAT MENGGUNAKAN *WIRELESS***

**Nama** : Moch Nafi Rizky Kurniawan

**Pembimbing** : Eko Pramunanto, ST., MT.

## **ABSTRAK**

Dewasa ini, *multi-display* sendiri digunakan untuk menampilkan suatu informasi. *Multi-display* disini disusun dari banyak layar televisi atau monitor untuk membentuk satu informasi yang utuh. Namun masih sedikit yang digunakan dalam ruang lingkup yang lebih kecil, yakni di sekolah atau di area perkantoran, karena di area tersebut juga membutuhkan suatu penyampaian informasi yang lebih efektif dan efisien.

Oleh karena itu, dibuatlah suatu tampilan *multi-display* dot matrix P10 yang dapat dikontrol secara terpusat sehingga dapat menampilkan informasi. Pengontrolan secara terpusat yang dilakukan disini adalah menggunakan suatu komputer ataupun laptop yang sudah terhubung dengan modul *wireless* transmitter yang terdiri atas Arduino dan Xbee yang dapat digunakan untuk mengirimkan informasi menuju tampilan *multi-display*. Sedangkan *multi-display* disini sendiri, terhubung dengan modul *wireless* receiver yang terdiri atas Arduino dan Xbee yang digunakan untuk menerima informasi yang dikirimkan oleh kontrol pusat. Jadi alat yang digunakan menggunakan metode *master* dan *slave* yang dikontrol secara terpusat dari jarak 100 meter untuk diluar ruang dan 20 meter pada didalam ruang. Alat ini dapat digunakan dalam rangka penyampaian informasi dari satu sumber informasi ke penerima secara *real time* sebanyak 63 karakter.

**Kata Kunci** : Arduino, *Multi-display* P10, Xbee

-----Halaman ini sengaja dikosongkan-----

# **CREATING OF A COMMUNICATION MODULE ON MULTI-DISPLAY WHICH IS CONTROLLED CENTRALLY USING WIRELESS**

**Nama : Moch Nafi Rizky Kurniawan**

**Pembimbing : Eko Pramunanto, ST., MT.**

## **ABSTRACT**

*Today, multi-display it self is used to display the information. Multi-display here composed of many television screen or monitor to form a complete information. But there is still little used in smaller scope, ie, at school or in the office area, because in these areas also need a peyampaian information more effectively and efficiently.*

*Therefore, they invented a multi-display P10 dot matrix display that can be centrally controlled so as to display the information. Centrally controlling done here is to use a computer or laptop that is connected to the wireless transmitter module consisting of Arduino and Xbee can be used to transmit information to the multi-view display. While the multi-display here alone, connected by wireless receiver module which consists of an Arduino and Xbee used for receiving information transmitted by the control center. So the tools used using the method of master and slave are controlled centrally from a distance of 100 meters for outside placedt and 20 meters on the inside place . This tool can be used in order of delivery information from one information source to the receiver in real time as many as 63 characters.*

**Keywords:** *Arduino, Multi-display P10, Xbee*

-----Halaman ini sengaja dikosongkan-----

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu diimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma 3 pada Bidang Studi Komputer Kontrol, Pogram Studi D3 Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

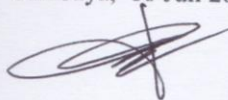
### **PEMBUATAN MODUL KOMUNIKASI PADA *MULTI-DISPLAY* YANG DIKONTROL SECARA TERPUSAT MENGGUNAKAN *WIRELESS***

Dalam Tugas Akhir ini *Multi-display* yang dikontrol secara terpusat menggunakan *wireless* yang berguna sebagai sarana penyampaian pengumuman pada instansi dalam ruang lingkup yang luas.

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Eko Pramunanto, ST., MT. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini, Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 18 Juli 2017



Penulis

-----Halaman ini sengaja dikosongkan-----

# DAFTAR ISI

	HALAMAN
HALAMAN JUDUL .....	i
PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
HALAMAN PENGESAHAN .....	vii
ABSTRAK .....	ix
<i>ABSTRACT</i> .....	xi
KATA PENGANTAR .....	xiii
DAFTAR ISI .....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xix
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Permasalahan .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Metodologi Penelitian .....	2
1.6 Sistematika Laporan .....	5
1.7 Relevansi .....	5
BAB II TEORI DASAR .....	7
2.1 Display P10 .....	7
2.1.1 IC APM4953 .....	9
2.2 Arduino <i>UNO</i> .....	10
2.3 IDE Arduino .....	12
2.4 XCTU .....	13
2.5 Xbee S2C .....	13
2.6 Power Supply .....	15
2.7 Modul Real-Time Clock (RTC) DS1307 .....	16
BAB III PERANCANGAN ALAT .....	21
3.1 Blok Fungsional Sistem .....	21
3.2 Perancangan Perangkat Keras .....	22
3.2.1 Kotak atau Bingkai <i>Multi-Display</i> .....	22
3.2.2 Perancangan <i>Multi-Display</i> (Modul DMD P10) .....	23
3.2.3 Modul Komunikasi pada <i>Slave</i> .....	25
3.2.4 Pembuatan Modul Komunikasi <i>Master</i> .....	28

3.3 Perancangan Perangkat Lunak .....	29
3.3.1 <i>Flowchart</i> .....	30
3.3.2 <i>Penyusunan Topology Jaringan Xbee</i> .....	33
3.3.3 <i>Setting Xbee dengan XCTU</i> .....	34
BAB IV HASIL IMPLEMENTASI ALAT .....	39
4.1 Pengujian <i>Multi-Display</i> .....	40
4.1.1 Pengujian Data yang Dikirim.....	40
4.1.2 Pengujian <i>Range</i> Komunikasi <i>Multi-Display</i> .....	42
4.2 Pengujian Keseluruhan Sistem.....	43
4.2.1 Hasil Pengiriman Data Komunikasi.....	44
4.2.2 Hasil Tampilan <i>Multi-Display</i> .....	44
4.2.3 Hasil Display Pengiriman dan Waktu Transmisi .....	45
4.2.4 Hasil Jarak yang Dapat Dijangkau oleh XBee S2C .....	46
BAB V PENUTUP .....	49
5.1 Kesimpulan .....	49
5.2 Saran .....	49
DAFTAR PUSTAKA .....	51
LAMPIRAN .....	55
A.1. Dokumentasi Pada Masa Percobaan Koneksi Xbee .....	55
A.2. Dokumentasi Proses Perakitan <i>Multi-Display</i> .....	58
A.3. Pengujian Data Xbee .....	61
A.4. Hasil Pengujian <i>RangeKomunikasi</i> .....	61
A.5 Dokumentasi Pengiriman Data dari PC ke <i>Multi-Display</i> .....	63
A.6. Listing Program .....	65
A.7. Datasheet .....	71
RIWAYAT HIDUP PENULIS.....	85



## DAFTAR GAMBAR

	HALAMAN
Gambar 2. 1 <i>Display</i> P10 Depan.....	8
Gambar 2. 2 <i>Display</i> P10 Tampak Belakang .....	8
Gambar 2. 3 <i>Display</i> P10 Konfigurasi .....	9
Gambar 2. 4 <i>Pinning</i> IC APM4953.....	10
Gambar 2. 5 Diagram Fungsional IC APM4953.....	10
Gambar 2. 6 Arduino Uno.....	11
Gambar 2. 7 <i>Software</i> IDE Arduino.....	12
Gambar 2. 8 <i>XCTU</i> .....	13
Gambar 2. 9 Xbee S2C .....	14
Gambar 2. 10 <i>Shield</i> Xbee <i>For</i> Arduino .....	14
Gambar 2. 11 Power Supply .....	15
Gambar 2. 12 Rangkaian Power Supply .....	16
Gambar 2. 13 Modul RTC .....	17
Gambar 2. 14 Modul RTC dengan Arduino.....	17
Gambar 2. 15 Koneksi Modul RTC .....	17
Gambar 2. 16 Modul RTC dan <i>Mikrokontroller</i> .....	19
Gambar 3. 1 Diagram Blok Fungsional Sistem.....	21
Gambar 3. 2 Desain Bingkai untuk <i>Multi-Display</i> .....	23
Gambar 3. 3 Hasil dari Rancangan Bingkai <i>Multi-Display</i> .....	23
Gambar 3. 4 Rangkaian Modul DMD P10 dengan Arduino .....	24
Gambar 3. 5 Sketsa <i>Multi-Display</i> Tampak Depan.....	25
Gambar 3. 6 Sketsa <i>Multi-Display</i> Tampak Belakang .....	25
Gambar 3. 7 Implementasi <i>Multi-Display</i> .....	25
Gambar 3. 8 Rangkaian Pin Xbee .....	26
Gambar 3. 9 Rangkaian Pin Xbee pada Arduino .....	27
Gambar 3. 10 Rangkaian <i>Shield</i> Pin Xbee pada Arduino .....	27
Gambar 3. 11 Implementasi Rangkaian Pin Xbee pada Arduino .....	28
Gambar 3. 12 Hasil Rancangan Modul Komunikasi <i>Master</i> .....	28
Gambar 3. 13 Hasil Modul Komunikasi pada <i>Master Slave</i> .....	29
Gambar 3. 14 Hasil Bingkai Modul Komunikasi <i>Master Slave</i> .....	29
Gambar 3. 15 <i>Flowchart</i> Transfer Data Xbee .....	31
Gambar 3. 16 <i>Flowchart</i> Program Penerimaan <i>Slave</i> .....	32
Gambar 3. 17 Konsep Dasar Jaringan Xbee .....	34

Gambar 3. 18 Konsep Jaringan <i>Mesh</i> Xbee.....	34
Gambar 3. 19 Proses Pembukaan dari Program <i>XCTU</i> .....	35
Gambar 3. 20 Halaman Awal dari Program <i>XCTU</i> .....	35
Gambar 3. 21 Halaman <i>Add Device</i> .....	36
Gambar 3. 22 Gambar Parameter Xbee .....	36
Gambar 4. 1 Hasil Slave Multi-Display.....	39
Gambar 4. 2 Hasil Modul Komunikasi Multi-Display.....	40
Gambar 4. 3 Pengujian dengan Xbee.....	41
Gambar 4. 4 Pengecekan dengan <i>XCTU</i> .....	44
Gambar 4. 5 Tampilan Multi-Display.....	45
Gambar 4. 6 Tampilan Pengumuman Multi-Display .....	45
Gambar 4. 7 Koneksi PC dan Modul Komunikasi.....	46
Gambar 4. 8 Tampilan <i>Multi-Display</i> di Dalam Ruang .....	47

## DAFTAR TABEL

	HALAMAN
Tabel 2. 1 Deskripsi Pin .....	11
Tabel 2. 2 Keterangan Fungsi Pin Modul Xbee .....	14
Tabel 2. 3 Fungsi Pin DS1307 .....	18
Tabel 4. 4 Konfigurasi Xbee pada XCTU .....	41
Tabel 4. 5 Data Pengiriman.....	41
Tabel 4. 6 Jarak Pengukuran Diluar Ruang.....	42
Tabel 4. 7 Jarak Pengukuran Didalam Ruang .....	43

-----Halaman ini sengaja dikosongkan-----

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Bidang teknologi dan komunikasi merupakan satu bidang dengan perkembangan yang cukup pesat dan terus menerus bertumbuh dari waktu ke waktu . salah satunya adalah teknologi komputer yang terus menerus menancapkan kiprahnya di dunia teknologi. Hal tersebut juga di iringi dengan berbagai fasilitas pendukung yang bisa di gunakan .

Komunikasi Data Menurut Suryadi H.S.,D (1998), komunikasi data adalah merupakan bagian dari teknologi komunikasi yang secara khusus berkenaan dengan transmisi atau pemindahan data dan informasi di antara komputer dan piranti-piranti yang lain dalam bentuk digital yang dikirimkan melalui media komunikasi data. Data berarti informasi yang disajikan oleh kode digital. Komunikasi data merupakan bagian penting dari suatu sistem informasi karena merupakan pendukung penyediaan infrastruktur yang memungkinkan komputer-komputer dapat berkomunikasi satu sama lain.

Saat ini, teknologi sudah memasuki era Jaringan nirkabel, Komunikasi nirkabel (*wireless*) adalah penyampaian informasi jarak jauh tanpa menggunakan konduktor listrik atau kawat dimana komunikasi nirkabel menggunakan media udara sebagai tempat transmisinya. Untuk dapat melakukan komunikasi secara *wireless*, sebuah *mikrokontroller* membutuhkan peralatan penerima dan juga pengirim. Jaringan *wireless* memiliki keunggulan dan keuntungan dibanding dengan jaringan kabel. Mobilitas jaringan *wireless* menyediakan pengaksesan kepada pengguna dimana saja, selama berada dalam batas aksesnya. Proses instalasi yang cepat dan mudah karena tidak membutuhkan kabel yang harus dipasang melalui atap atau tembok. Fleksibilitas tempat jaringan *wireless* sangat fleksibel terhadap tempat berbeda dengan jaringan kabel yang dipasang tanpa kabel. Pengurangan anggaran biaya terjadi saat terdapat perpindahan tempat walaupun investasi awal pada *wireless* lebih besar daripada jaringan kabel. Biaya instalasi dapat diperkecil karena tidak membutuhkan kabel dan biaya pemeliharaan yang lebih murah. Kemampuan jangkauan konfigurasi jaringan dapat diubah

dari jaringan *peer-to-peer* untuk jumlah pengguna yang sedikit menjadi jaringan infrastruktur yang banyak hingga mencapai ribuan pengguna yang dapat menjelajah dengan jangkauan luas. Jangkauan kontroler yang cukup luas yang dapat mengontrol beberapa pun *plant* di rasa sangat membantu sekali.

## **1.2 Permasalahan**

Panjangnya dan banyaknya kabel yang digunakan pada sistem konvensional yang akan membutuhkan banyak ruang sehingga memakan tempat dan akan menimbulkan ketidak-praktisan dan tidak efisien dalam penataan dan kenyamanan dalam ruangan. Sehingga dapat mengganggu dalam proses perawatan dan perbaikan apabila terjadi *trouble* atau masalah. Alat yang akan kami buat, akan menyampaikan informasi secara langsung tanpa menggunakan kabel yang di mana akan menggunakan bantuan komunikasi *wireless* yang memungkinkan upaya meminimalisir kabel dan menambah daya jangkau informasinya.

## **1.3 Batasan Masalah**

Dalam pembuatan alat pada Tugas Akhir ini batasan masalah ada pada *display* yang di kontrol secara terpusat yang hanya dapat menyampaikan informasi secara visual berbentuk teks.

## **1.4 Tujuan**

Tujuan dari menuliskan Tugas Akhir ini adalah menyampaikan informasi yang dibutuhkan melalui *multi-display dot matrix* dengan komunikasi *wireless* (nirkabel/tanpa kabel) secara terpusat sesuai dengan yang diinginkan.

## **1.5 Metodologi Penelitian**

Dalam pelaksanaan Tugas Akhir yang berupa *multi-display* yang dikontrol secara terpusat menggunakan *wireless*, ada beberapa kegiatan yang dapat diuraikan sebagai berikut:

### **1. Study Pustaka dan Survei**

Pengumpulan data-data dari berbagai literatur mendukung yang dalam proses pembuatan Tugas Akhir ini. Setelah itu melakukan survei di mana alat kami dapat di pasang pada jurusan Departemen Teknik Elektro otomasi. Dan menunjukkan berbagai keunggulan dari produk.

## **2. Tahap Persiapan**

Pada tahap ini akan dilakukan studi literatur mengenai:

- a. Mempelajari macam-macam *Topology* Xbee
- b. Mempelajari konsep dasar Xbee *programing* dan konfigurasi.
- c. Mempelajari sistem Xbee
- d. Mempelajari proses pengiriman data dari Xbee ke masing-masing *Multi-display*.

## **3. Tahap Identifikasi dan Pemodelan Sistem**

Pada tahap ini akan dilakukan identifikasi dari sistem alat sesuai data yang telah didapatkan dari studi literatur serta dilakukan pemodelan dari alat yang akan dikerjakan.

## **4. Tahap Perancangan Alat**

### **a. Object Penelitian**

Sesuai dengan tujuan Tugas Akhir yang telah diuraikan maka objek penelitian dalam penyusunan Tugas Akhir ini meliputi :

- a. Proses Jaringan nirkabel (*wireless*) Xbee.
- b. pengiriman data pada *multi-display*.

### **b. Tempat Perancangan**

Untuk melakukan kegiatan Tugas Akhir ini akan melakukan penelitian yang akan dilakukan di:

- a. Lab BB 102 Laboratorium Elektronika dasar.
- b. Lab BB 202 *Automation Computer Laboratory*.
- c. Rumah Pribadi di jalan medokan semampir indah 117.
- d. Rumah Kos di jalan Keputih gang 2B No.25.
- e. Lab BB 104 Laboratorium Elektronika Terapan.

### **c. Waktu Perancangan**

Dalam melakukan kegiatan Tugas Akhir ini akan melakukan penelitian Mulai dari Bulan Februari 2017 sampai dengan Bulan Juni 2017.

### **d. Alat Terkait**

Dalam melakukan Tugas Akhir ini kami menggunakan beberapa alat di antaranya sebagai berikut:

- a. Xbee S2C
- b. Arduino Uno R3

- c. *Display DMD P10*
- d. *Real Time Clock(RTC)*
- e. *power supply switching 5V 10A*

## 5. Tahap Pembuatan Alat

Perancangan *hardware* yang dilakukan dengan merancang rangkaian-rangkaian penyusun *multi-display* yang akan di gunakan dalam masing-masing *slave* meliputi:

- a. Konfigurasi Arduino dan Xbee.
- b. Rancangan Tata letak dan power dari panel.

Perancangan *Software* meliputi.

- a. Menyusun algoritma untuk mengkoordinasikan kerja masing-masing komponen.
- b. Menyusun proses *scanning* dan juga penyampaian informasi melalui *wireless* dengan melakukan *setting input*.
- c. Penyusunan *flowchart* untuk *slave* dan proses pengiriman data dari modul Xbee.

## 6. Tahap Pengujian dan Analisa

Dalam proses uji coba dan analisis data terhadap alat diharapkan akan membantu dan mempermudah mahasiswa apabila alatnya mengalami kesalahan atau kegagalan. Tahapan ini dilakukan dengan cara melakukan pengecekan alat. Setelah dilakukan, melakukan analisis terhadap kesalahan dan kegagalan apabila terjadi, lalu dilakukan pemecahan masalah terhadap permasalahan tersebut. Beberapa pengujian yang akan dilakukan adalah sebagai berikut.:

- a. Ketepatan pengiriman dan penerimaan data melalui Xbee dari *master* ke beberapa *slave*.
- b. Ketepatan informasi yang ditampilkan oleh *multi-display P10*.
- c. Jangkauan jarak yang dapat dijangkau untuk pengiriman dan penerimaan data dari *master* ke beberapa *slave*.

## 7. Tahap Penyusunan Laporan

Setelah alat berhasil dibuat dan berkerja dengan baik tanpa adanya *error*, pengambilan data dan analisa data terpenuhi, maka tahap selanjutnya yaitu penyusunan laporan untuk buku Tugas Akhir. Diharapkan buku Tugas Akhir ini bermanfaat bagi semua orang, dan



dapat dijadikan pedoman dalam melanjutkan dan mengembangkan ide Tugas Akhir ini.

### **1.6 Sistematika Laporan**

Untuk pembahasan lebih lanjut, laporan Tugas Akhir ini disusun dengan sistematika sebagai berikut:

#### **Bab I PENDAHULUAN**

Membahas tentang latar belakang, perumusan masalah, batasan masalah, maksud dan tujuan, sistematika laporan, metodologi, serta relevansi Tugas Akhir yang dibuat.

#### **Bab II TEORI DASAR**

Menjelaskan teori yang berisi teori-teori dasar yang dijadikan landasan dan mendukung dalam perencanaan dan pembuatan alat yang dibuat.

#### **Bab III PERANCANGAN ALAT**

Membahas perencanaan dan pembuatan tentang perencanaan dan pembuatan *hardware* yang meliputi desain mekanik dan perancangan *software* yang meliputi program yang akan digunakan untuk menjalankan alat tersebut.

#### **Bab IV PENGUKURAN DAN ANALISA**

Membahas pengujian alat dan menganalisa data yang didapat dari pengujian tersebut serta membahas tentang pengukuran, pengujian, dan penganalisaan terhadap alat.

#### **Bab V PENUTUP**

Berisi penutup yang menjelaskan tentang kesimpulan yang didapat dari Tugas Akhir ini dan saran-saran untuk pengembangan alat ini lebih lanjut.

### **1.7 Relevansi**

Dari pembuatan alat ini diharapkan akan tercipta beberapa manfaat yaitu:

- a. Mempermudah pengontrolan *multi-display* dari jarak jauh.
- b. Mempermudah penyaluran informasi.
- c. Dapat digunakan dalam radius yang cukup jauh (tergantung dari kemampuan suatu komponen *wireless* yang digunakan).
- d. Memperoleh data informasi yang akurat dan terpercaya.

## BAB II

### TEORI DASAR

Beberapa teori penunjang yang dipaparkan dalam buku Tugas Akhir ini adalah teori dasar mengenai proses penulisan dan juga proses kontrol pengiriman data dengan menggunakan Xbee sebagai sarana penghubung secara nirkabel. Karena komunikasi tanpa kabel atau nirkabel (*wireless*) telah menjadi kebutuhan dasar atau gaya hidup baru masyarakat. informasi nirkabel yang lebih dikenal dengan jaringan *wireless* menjadi teknologi alternatif dan *relative* mudah diimplementasikan dilingkungan kerja seperti perkantoran, laboratorium komputer, dan sebagainya. Instalasi perangkat jaringan *wireless* lebih fleksibel karena tidak membutuhkan penghubung kabel antar komputer. komputer dengan *Wireless device* dapat saling terhubung yang hanya membutuhkan ruang atau space dengan syarat jarak jangkauan dibatasi kekuatan pancaran sinyal radio dari masing-masing komputer. (Priyambodo, Tri Kuntoro & Heriadi, Dedi. 2005 :1) maka dapat diterapkan teknologi *wireless* ini untuk kebutuhan manusia agar saling berkomunikasi dengan memanfaatkan udara sebagai medium penghantar sinyalnya.

#### 2.1 Display P10

*Dot matrix display* (sering disingkat menjadi istilah DMD) Adalah sebuah LED *display* yang dirancang dengan ukuran 16x64 yang dapat digunakan dalam luar ruang maupun dalam ruang. Pada *display* ini dapat disambungkan dengan *display* P10 lainnya dengan paralel maupun seri. Pada display ini dapat digunakan masukkan dari *power supply* maupun dari Arduino langsung yang terhubung dengan PC. Penggunaan *display* ini banyak digunakan untuk *running led*. *Running text* adalah papan iklan promosi dan informasi yang terbuat dari LED yang dapat diprogram menggunakan komputer untuk menampilkan tulisan, gambar dan animasi berwarna. Pada umumnya, *running text* dibedakan menjadi dua, yaitu *running text outdoor* dan *indoor*. *Running text outdoor* adalah *running text* yang diletakkan diluar ruangan dan biasanya menggunakan jenis LED DIP (bentuk *led* yang benjol-benjol) Seperti **Gambar 2.1** dan **Gambar 2.2** , sedangkan *running text indoor* adalah *running text* yang

diletakkan di dalam ruangan dan biasanya menggunakan jenis LED *dot matrix* (bentuk *led* yang permukaannya datar).

Penamaan P10 pada *display* ini maksudnya adalah jarak dari led ke led yang lain ada 10 mm. Ini didasarkan pada arti dari Pxx sendiri adalah *Pixel Pitch*. Maka dari itu, ada banyak sekali tipe dari *display* ini diantaranya P10, P16, P8, P3, dan lain-lain.



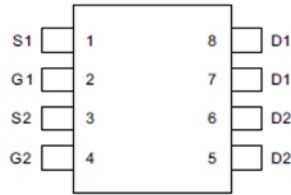
**Gambar 2. 1** *Display* P10 Depan



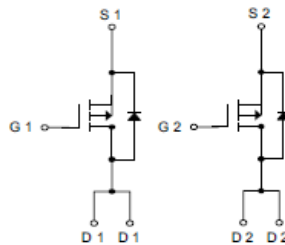
**Gambar 2. 2** *Display* P10 Tampak Belakang

*Display* dapat disusun menjadi 4 dimana terdiri dari 2 kolom dan 2 baris. Digunakan modul led P10 pada tugas akhir ini dikarenakan modul ini sesuai dengan apa yang diharapkan oleh penulis, yaitu mempunyai dimensi 32x16, tidak terlalu besar dan tidak terlalu kecil, dan juga memudahkan dalam proses pengkoneksian atau penyambungannya dengan modul P10 yang lain. Ini dapat dijelaskan dari rangkaian dalam modul P10 ini sebagai berikut **Gambar 2.3.**





**Gambar 2. 4** Pinning IC APM4953



**Gambar 2. 5** Diagram Fungsional IC APM4953

## 2.2 Arduino UNO

Dalam buku “*Getting started with Arduino*” Arduino adalah sebuah *platform* komputasi fisik yang *open source* pada *board input output* sederhana. Yang dimaksud dengan *platform* komputasi fisik di sini adalah sebuah sistem fisik yang interaktif dengan penggunaan *software* dan *hardware* yang dapat mendeteksi dan merespon situasi dan kondisi yang ada di dunia nyata.

Arduino UNO adalah sebuah *board mikrocontroller* yang didasarkan pada ATmega328. Arduino UNO mempunyai 14 pin digital input/output (6 di antaranya dapat digunakan sebagai output PWM), 6 input analog, sebuah osilator kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah ICSP header, dan sebuah tombol *reset*.

Arduino Uno (**Gambar 2.6**) adalah alat yang diperlukan untuk mendukung sebuah rangkaian *mikrokontroler*. Cukup dengan menghubungkannya ke komputer dengan kabel USB atau diberi *power* dengan adaptor AC-DC atau baterai, Arduino uno sudah dapat digunakan tanpa khawatir akan melakukan sesuatu yang salah.



**Gambar 2. 6** Arduino Uno

Untuk meringkasnya, diberikan gambar dibawah sebagai ringkasan dari penjelasan Arduino UNO pada **Tabel 2.1**.

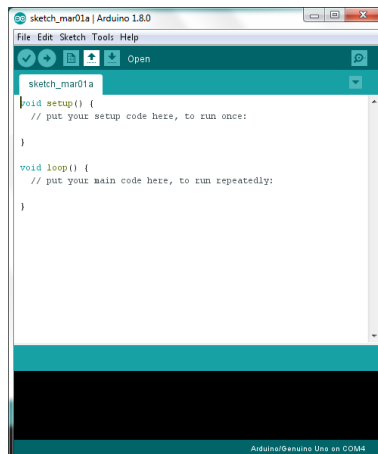
**Tabel 2. 1** Deskripsi Pin

<b>Mikrokontroler</b>	<b>ATmega328</b>
Tegangan pengoperasian	5 Volt
Tegangan Input yang di sarankan	7-12V
Batas tegangan input	5-20V
Jumlah pin I/O digital	14 (6 diantaranya Untuk PWM)
Jumlah pin input analog	6
Arus dc tiap pin I/O	40 mA
Arus DC untuk pin 3,3 V	50 mA
Memori Flash	32 KB (Atmega328) , 0,5 bootloader
SRAM	2 KB(Atmega 328)
EEPROM	1 KB(Atmega328)
Clock speed	16 Mhz

“Uno” berarti satu dalam bahasa Italia dan dinamai untuk menandakan keluaran (produk) Arduino 1.0 selanjutnya. Arduino UNO dan versi 1.0 akan menjadi referensi untuk versi-versi Arduino selanjutnya. Arduino UNO adalah sebuah seri terakhir dari *board* Arduino USB dan model referensi untuk papan Arduino, untuk suatu perbandingan dengan versi sebelumnya.

### 2.3 IDE Arduino

IDE itu merupakan kependekan dari *Integrated Development Enviroenment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui *software* inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dinamakan melalui sintaks pemrograman seperti **Gambar 2.7**. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC *mikrokontroller* Arduino telah ditanamkan suatu program bernama *bootlader* yang berfungsi sebagai penengah antara *compiler* Arduino dengan *mikrokontroller*. Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan library C/C++ yang biasa disebut *wiring* yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari *software processing* yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino.

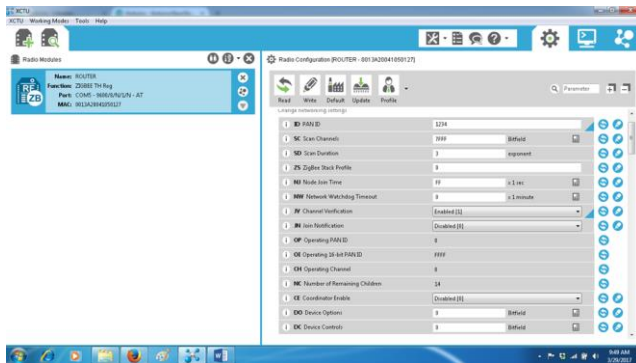


**Gambar 2. 7** Software IDE Arduino



## 2.4 XCTU

XCTU adalah Sebuah program yang di mana kita dapat mengatur sendiri konfigurasi dari Xbee agar memiliki ruang kerja yang sama. XCTU sendiri bisa juga di gunakan sebagai program untuk sarana Xbee berkomunikasi secara *wireless* tanpa perlu menggunakan *board* lain sebagai media. Pada program XCTU menyediakan fitur agar kita dapat mengecek secara langsung untuk proses data yang di kirim menggunakan Xbee sudah sesuai apa belum. Pada dasarnya XCTU untuk mengkonfigurasi dan Menentukan ruang kerja Xbee agar memiliki satu alamat sama. Untuk mengetahui lebih lanjut tentang XCTU bisa di lihat pada **Gambar 2.8**.



**Gambar 2. 8** *XCTU*

## 2.5 Xbee S2C

Arduino Xbee teknologi yang dikembangkan untuk memantau dan merekam parameter. Sistem ini terdiri dari PIC mikrokontroler untuk penanganan sensor dan sinyal dikomunikasikan melalui Xbee *transmitter* dan Xbee *receiver*. Xbee berbasis nirkabel, perangkat ini digunakan.

Sebagai komunikasi nirkabel jarak antara modul. Informasi yang ditransmisikan titik dengan titik menggunakan Xbee *transmitter* dan *receiver* dan dikirim ke *server* modul untuk memeriksa keadaan parameter. **Tabel 2.2** menjelaskan tentang pin dari Xbee dan **Gambar 2.9** adalah perangkat Xbee.

**Tabel 2. 2 Keterangan Fungsi Pin Modul Xbee**

Pin	Name	Direction	Description
1	VCC	-	Power Supply
2	DOUT	Output	UART Data Out
3	DIN/CONFIG	Input	UART Data In
5	Reset	Input	Modul Reset
6	PWM0 / RSSI	Output	PWM Output 0/RX Signal Indicator
10	GND	-	Ground



**Gambar 2. 9** Xbee S2C

Untuk *shield* dari Xbee ini sendiri memiliki konfigurasi sama dengan *board* Arduino uno di karenakan untuk mengatasi input tegangan dan juga pin TR RX nya. Untuk *shield* di tunjukkan pada **Gambar 2.10**.



**Gambar 2. 10** Shield Xbee For Arduino

## 2.6 Power Supply

Pada *Power Supply* atau dalam bahasa Indonesia disebut dengan Catu Daya adalah suatu alat listrik yang dapat menyediakan energi listrik untuk perangkat listrik ataupun elektronika lainnya. Pada dasarnya *Power Supply* atau Catu Daya ini memerlukan sumber energi listrik yang kemudian mengubahnya menjadi energi listrik yang dibutuhkan oleh perangkat elektronika lainnya. Oleh karena itu, *Power Supply* kadang-kadang disebut juga dengan istilah *Electric Power Converter*.

*Power Supply* **Gambar 2.11** dapat dibedakan menjadi beberapa jenis, diantaranya adalah *DC Power Supply*, *AC Power Supply*, *Switch-Mode Power Supply*, *Programmable Power Supply*, *Uninterruptible Power Supply*, *High Voltage Power Supply*. Yang digunakan pada tugas ini adalah *Switch-Mode Power Supply*. Dimana, *Switch-Mode Power Supply* adalah jenis *Power Supply* yang langsung menyearahkan (*rectify*) dan menyaring (*filter*) tegangan *input* AC untuk mendapatkan tegangan DC. Tegangan DC tersebut kemudian di-switch ON dan OFF pada frekuensi tinggi dengan sirkuit frekuensi tinggi sehingga menghasilkan arus AC yang dapat melewati *Transformator* Frekuensi Tinggi. Untuk rangkaian dari *power supply* ditunjukkan pada **Gambar 2.12**.



**Gambar 2. 11** Power Supply



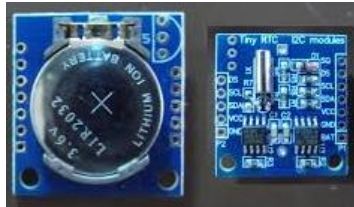
**Gambar 2. 12** Rangkaian Power Supply

## 2.7 Modul Real-Time Clock (RTC) DS1307

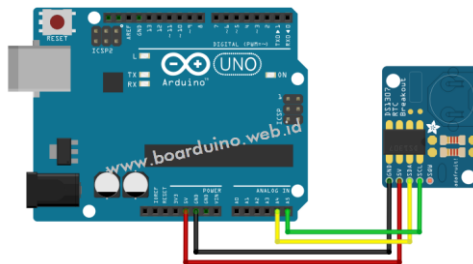
Modul *Real-time clock* disingkat RTC **Gambar 2.13** adalah jam di komputer yang umumnya berupa sirkuit terpadu yang berfungsi sebagai pemelihara waktu. RTC umumnya memiliki catu daya komputer (umumnya berupa baterai litium) sehingga dapat tetap berfungsi ketika catu daya komputer terputus. Kebanyakan RTC menggunakan *oscillator* kristal dan mudah di koneksikan dengan Arduino seperti **Gambar 2.14**.

RTC tipe DS1307 merupakan jenis pewaktu yang menggunakan komunikasi serial untuk operasi tulis baca, dengan spesifikasi berikut ini:

- *Real-time clock* (RTC) menyimpan data-data detik, menit, jam, tanggal, bulan, hari dalam seminggu, dan tahun dimana valid hingga 2100.
- 56-byte, *battery-backed*, *RAM nonvolatile* (NV) RAM untuk penyimpanan.
- Antarmuka serial *Two-wire* (I2C).
- Sinyal luaran gelombang-kotak terprogram (*Programmable squareware*).
- Deteksi otomatis kegagalan-daya (*power-fail*) dan rangkaian *switch*.
- Konsumsi daya kurang dari 500 nA menggunakan mode baterai cadangan dengan operasional *oscillator*.
- Tersedia fitur industri dengan ketahanan suhu: -40°C hingga +85°C.
- Tersedia dalam kemasan 8-pin DIP atau SOI.

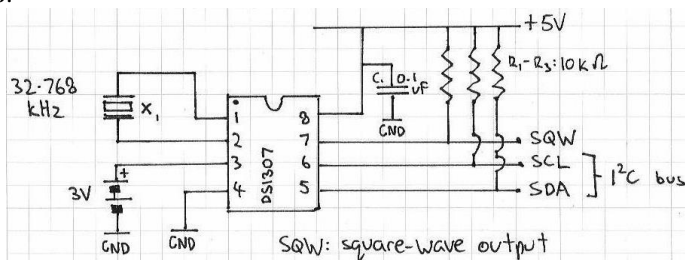


**Gambar 2. 13** Modul RTC



**Gambar 2. 14** Modul RTC dengan Arduino

Sistem RTC DS1307 memerlukan baterai eksternal 3V yang terhubung ke pin Vbat dan *ground*. Pin X1 dan X2 dihubungkan dengan kristal osilator 32,768 KHz. Sedangkan pin SCL, SDA, dan SQW/OUT di-*pull-up* dengan resistor (nilainya 1k s.d 10k) ke Vcc. Skema lengkapnya pada **Gambar 2.15** dan untuk koneksi pin pada **Gambar 2.16**:



**Gambar 2. 15** Koneksi Modul RTC

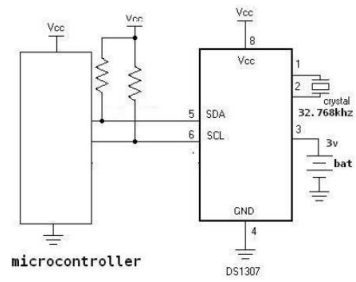
**Tabel 2. 3** Fungsi Pin DS1307

No.	Nama Pin	Fungsi
1	X1	<i>Oscillator</i> Crystal 32.768 KHz
2	X2	
3	VBAT	Battery Input (+3V)
4	GND	Ground
5	SDA	Serial Data
6	SCL	Serial Clock Input
7	SQW/OUT	Square Wave/Output Driver
8	VCC	Supply Power

Keterangan:

- Pin Vcc (Nomor 8) berfungsi sebagai sumber energi listrik utama. Tegangan kerja dari komponen ini adalah 5V, dan ini sesuai dengan tegangan kerja dari *mikrokontroller* papan Arduino.
- Pin GND (Nomor 4) harus dihubungkan *ground* yang dimiliki oleh komponen RTC dengan *ground* dari *battery back-up*.
- SCL berfungsi sebagai saluran *clock* untuk komunikasi data antara *mikrokontroller* dengan RTC.
- SDA berfungsi sebagai saluran data untuk komunikasi data antara *mikrokontroller* dengan RTC.
- X1 dan X2 berfungsi untuk saluran *clock* yang bersumber dari *crystal external*.

Vbat berfungsi sebagai saluran energi listrik dari *battery external*. Bentuk komunikasi data dari IC RTC adalah I2C yang merupakan kepanjangan dari *Inter-Integrated Circuit*. Komunikasi jenis ini hanya menggunakan 2 jalur komunikasi, yaitu SCL dan SDA. Semua mikrokontroler sudah dilengkapi dengan fitur komunikasi 2 jalur ini, termasuk diantaranya *Mikrokontroller* Arduino seperti **Gambar 2.16**.



**Gambar 2. 16** Modul RTC dan *Mikrokontroller*

-----Halaman ini sengaja dikosongkan-----

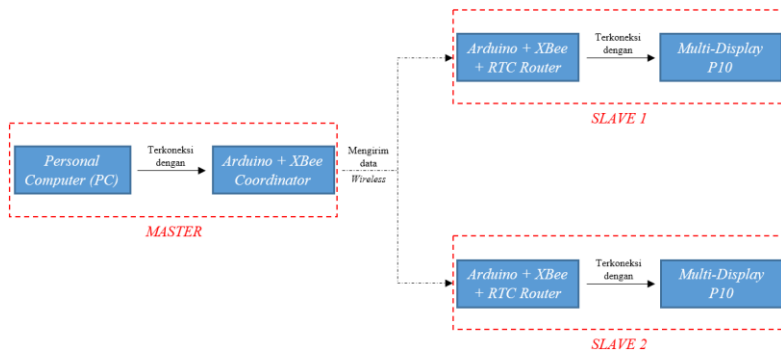


## BAB III

### PERANCANGAN ALAT

Pada Bab ini akan dijelaskan mengenai perancangan dan pembuatan Sistem *Multi-display* yang di kontrol terpusat menggunakan *Wireless* meliputi, pengenalan Xbee S2C , blok fungsional sistem yang akan menjelaskan proses kerja alat dalam bentuk alur diagram, Perancangan *Hardware* meliputi Perancangan *multi-display* , Perancangan RTC ,Perancangan kabel daat dari Arduino, Xbee S2C dan *Shield*. , Power Supply , Pembuatan bingkai dan untuk perancangan *software* menjelaskan tentang *Flowchart* alur kerja alat dan juga menjelaskan pengaturan dari Setting Xbee dengan XCTU.

#### 3.1 Blok Fungsional Sistem



**Gambar 3. 1** Diagram Blok Fungsional Sistem

Dari **Gambar 3. 1**, dijelaskan bahwa dalam Tugas Akhir ini, akan membuat sebuah *multi-display* yang di mana dalam satu panel terdiri dari empat *dot matriks display P10*. Yang kami kemas dengan metode bingkai aluminium yang sesuai dengan ukuran dari *dot matriks*. Pada bagian bingkai kami rancang dengan kuat agar tidak mudah goyah dan juga dapat menahan apabila terjadi benturan yang dapat membahayakan komponen-komponen penyusun dalam panel *multi-display*. Pada masing-masing dari *dot matrix display* menerima masukkan dari *power supply* secara seri terbagi pada bagian atas sebesar 5 volt dan juga pada

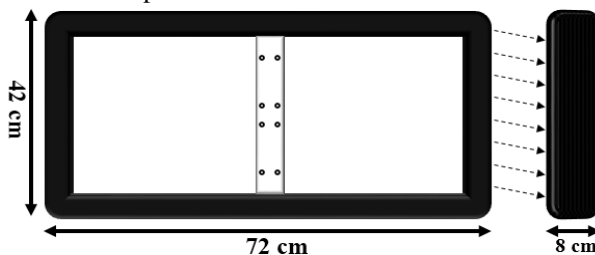
bagian bawah. Pada perancangan dari tugas akhir ini kami menggunakan sistem *scanning* secara terurut yang dimana kami menggunakan Xbee sebagai media transmisi nirkabel (*wireless*) adalah penyampaian informasi jarak jauh tanpa menggunakan konduktor listrik atau kawat. Xbee membutuhkan peralatan penerima dan juga pengirim. Jaringan *wireless* memiliki keunggulan dan keuntungan dibanding dengan jaringan kabel. Mobilitas jaringan *wireless* menyediakan akses ke pengguna dimana saja, selama berada dalam batas aksesnya. yang dapat di kontrol dengan jarak jauh. Kita selaku operator dari *multi-display* bisa mengatur pengumuman yang akan kita sampaikan pada *multi-display*. Kita memiliki fitur untuk keadaan *stanby* maka *display* akan mengeluarkan *text static* yang akan menunjukkan tempat dan juga waktu *display* itu berada. Pada alat kami pula pengumuman dapat di kirim secara *real time* yang akan sangat menguntungkan.

### 3.2 Perancangan Perangkat Keras

Pada Bab ini akan dibahas tentang perancangan rangkaian pengkondisi sinyal dan perhitungan komponen pendukungnya, beserta *wiring* dari *hardware* yang digunakan. Secara keseluruhan rangkaian akan di jelaskan seperti di bawah ini .

#### 3.2.1 Kotak atau Bingkai *Multi-Display*

Agar penampilan dari *multi-display* menjadi lebih rapi dan mempunyai bentuk yang tetap, maka dirancanglah suatu kotak atau bingkai yang digunakan untuk menempatkan *multi-display*. Bingkai akan dibuat dari bahas aluminium, karena lebih ringan dan fleksibel ketika ingin dibawa kemana saja. Desain dari rancangan pembuatan bingkai ini diberikan pada **Gambar 3.2**.



**Gambar 3. 2** Desain Bingkai untuk *Multi-Display*

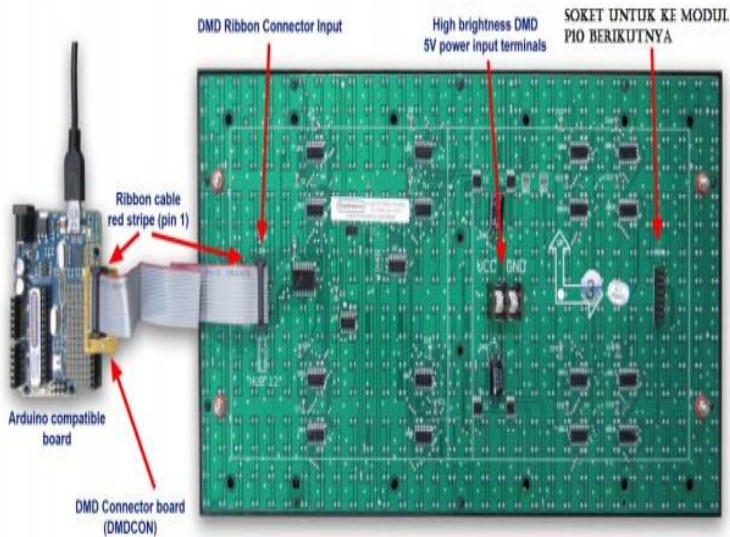
Pada gambar di atas adalah desain bingkai untuk meletakkan panel P10, atau *multi-display*, dengan ukuran yaitu panjang 72 cm, lebar 42 cm, dan tinggi 8 cm. Pada bagian sisi juga diberikan bagian yang diberi nama siku untuk memperkuat dari desain ini. Bagian tengah dari desain ini diberikan sebuah *din rail* sebagai penopang dari modul-modul DMD P10 yang akan dipasang. Realisasi dari desain tersebut pada **Gambar 3.3.**



**Gambar 3. 3** Hasil dari Rancangan Bingkai *Multi-Display*

**3.2.2 Perancangan *Multi-Display* (Modul DMD P10)**

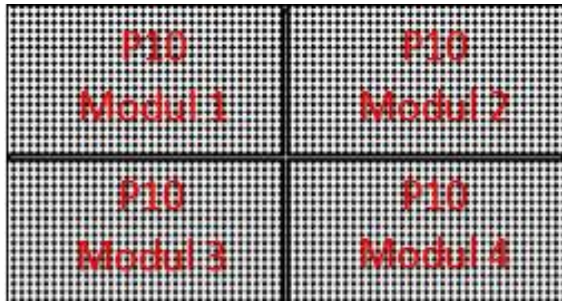
Perancangan *multi-display* ini tersusun atas 4 modul DMD P10 dan bingkai atau kotak untuk menempatkan modul P10 tersebut. Seperti yang telah dijelaskan diatas, perancangan *multi-display* disini menggunakan modul DMD yang diberi nama modul P10. Penamaan P10 pada *display* ini maksudnya adalah jarak dari *led* ke *led* yang lain ada 10 mm. Ini didasarkan pada arti dari Pxx sendiri adalah *Pixel Pitch*. Maka dari itu, ada banyak sekali tipe dari *display* ini diantaranya P10, P16, P8, P3,75, dan lain-lain. Modul ini dapat disusun menjadi 4 dimana terdiri dari 2 kolom dan 2 baris. Digunakannya modul DMD P10 pada Tugas Akhir ini dikarenakan modul ini sesuai dengan apa yang diharapkan oleh penulis, yaitu mempunyai dimensi 32x16, tidak terlalu besar dan tidak terlalu kecil, dan juga dimudahkan dalam proses peng-koneksi-an atau penyambungannya dengan modul P10 yang lain. Ini dapat dijelaskan dari rangkaian dalam modul P10 pada **Gambar 3.4.**



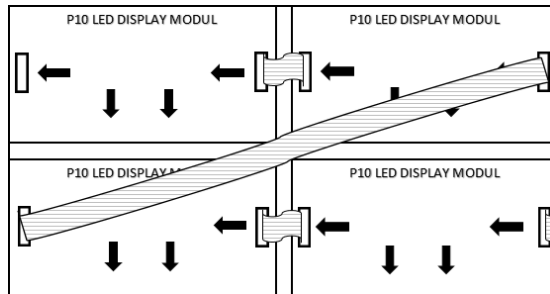
**Gambar 3. 4** Rangkaian Modul DMD P10 dengan Arduino

Penjelasannya, data akan dikirim dari Arduino secara serial melalui soket HUB1. kemudian diterima oleh IC *serial-to-parallel* 74595. Jika ada tambahan modul akan diambil dari keluaran 74595 yang terakhir yang dihubungkan ke soket X2-OUT. Untuk *supply* arus diberikan oleh IC *driver* penguat daya VT1,VT2..dst. Untuk memilih kolom mana yang menyala diaktifkan oleh IC *demultiplexer*. Jika digambarkan dalam bentuk modul sudah jadi atau siap pakai, maka susunannya adalah sebagai berikut.

Dari gambar diatas, secara sederhana, jika ingin menambahkan modul P10 lagi, dapat ditambahkan melalui “*data output*” led modul pertama ke “*data input*” led modul kedua dan seterusnya menggunakan kabel *ribbon*. Sehingga, jika digambarkan sketsanya maka akan membentuk sebagai berikut **Gambar 3.5** dan **Gambar 3.6**. untuk implementasinya pada **Gambar 3.7**.



**Gambar 3. 5** Sketsa *Multi-Display* Tampak Depan



**Gambar 3. 6** Sketsa *Multi-Display* Tampak Belakang

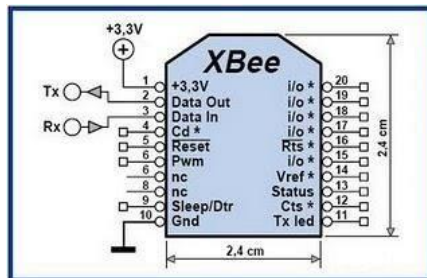


**Gambar 3. 7** Implementasi *Multi-Display*

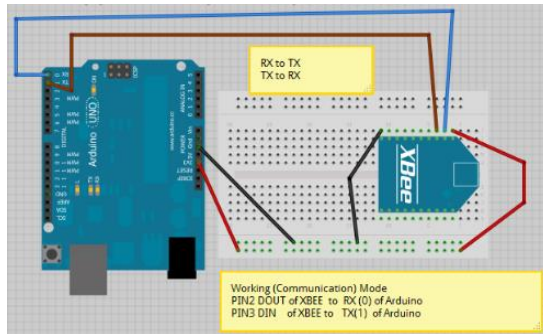
### 3.2.3 Modul Komunikasi pada *Slave*

Untuk rangkaian dari *shield* Xbee pada Arduino hanya menggunakan beberapa pin saja yaitu pin dari Vcc dan *ground* untuk mengaktifkan Xbee agar menyala ,dan pin data yaitu TR, RX yang di mana akan berguna dalam pengiriman dan penerimaan data secara

*wireless* tanpa adanya kabel. Dalam mengintegrasikan Xbee dengan Arduino, perlu diperhatikan Perangkat ini memiliki 20 pin dengan fungsi yang berbeda-beda. Untuk koneksi minimum, dibutuhkan pin VCC, GND, DOUT & DIN. Sedangkan untuk dapat melakukan update *firmware*, dibutuhkan koneksi pin VCC, GND, DIN, DOUT, RTS & DTR. VCC dan GND untuk tegangan suplai, DOUT merupakan pin Transmit (TX), DIN merupakan pin *Receive* (RX), RESET merupakan pin reset Xbee PRO dan yang terakhir adalah PWMO/RSSI merupakan indikator bahwa ada penerimaan data yang biasanya dihubungkan ke led yang di-*drive* oleh transistor. Untuk mengaktifkan Xbee dibutuhkan supply tegangan sebesar 3.3 V. Xbee merupakan modul *transceiver*, dengan mekanisme pengiriman data secara serial. Pin-pin untuk mengirim dan menerima data ada di pin 2 sebagai pin data OUT (Tx) dan pin 3 sebagai pin data IN (Rx). sehingga sangat penting untuk mengetahui pin Tx dan Rx. Bentuk fisik dan konfigurasi pin Xbee S2C dengan Arduino ditunjukkan pada **Gambar 3.8** dan **Gambar 3.9**.



**Gambar 3. 8** Rangkaian Pin Xbee



**Gambar 3. 9** Rangkaian Pin Xbee pada Arduino

*Connector* dengan Arduino menggunakan *shield* Xbee (**Gambar 3.10**) yang berguna dalam menjadi perantara antara Xbee dan Arduino tanpa perlu menghawatirkan terjadinya kelebihan tegangan yang dapat merusak Xbee itu sendiri.



**Gambar 3. 10** Rangkaian *Shield* Pin Xbee pada Arduino

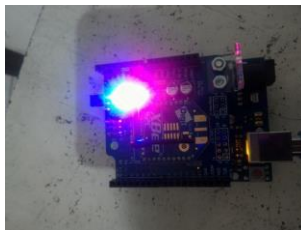


**Gambar 3. 11** Implementasi Rangkaian Pin Xbee pada Arduino

Dari yang di tunjukkan oleh **Gambar 3.11.** Adalah hasil dari pembuatan modul komunikasi pada slave. Untuk modul komunikasi tersebut langsung terkoneksi dengan rangkaian RTC dan juga rangkaian Kabel data yang akan masuk ke dalam *display* P10.

#### **3.2.4 Pembuatan Modul Komunikasi *Master***

Untuk modul komunikasi sendiri menggunakan Xbee yang di sambungkan dengan modul perantara ke Arduino atau biasa di sebut dengan *Shield* Xbee ke Arduino. Pada *Shield* ini sudah menghubungkan antara VCC , *Ground* , Pin TR ,Pin RX dari Xbee ke modul Arduino.

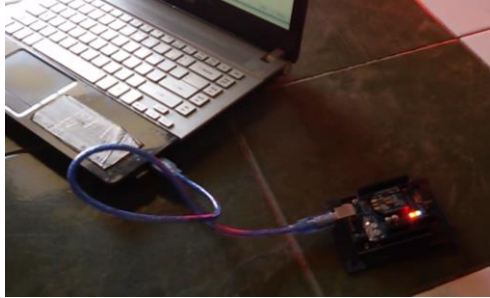


**Gambar 3. 12** Hasil Rancangan Modul Komunikasi *Master*

Setelah membuat modul untuk *master* **Gambar 3.12** dan **Gambar 3.13.** lanjutkan untuk membuat kerangka pelindung dari modul *master* yang akan di gunakan dalam PC *master*. Pada proses ini di butuhkan



sebuah kotak yang di mana sudah memiliki ukuran yang sesuai untuk tempat dari modul *master*.



**Gambar 3. 13** Hasil Modul Komunikasi pada *Master Slave*

Jadi setelah selesai menghitung ukuran dari *master slave*. lakukan proses pengepakan dan hingga module *master slave* tertutup dengan rapi Seperti **Gambar 3.14**.



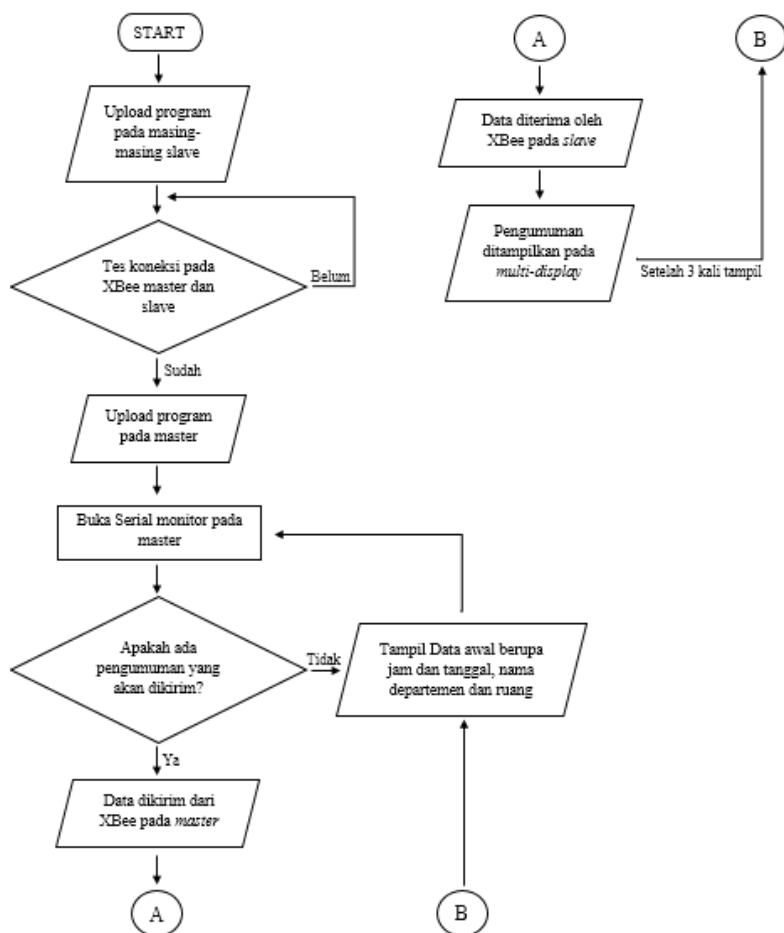
**Gambar 3. 14** Hasil Bingkai Modul Komunikasi *Master Slave*

### **3.3 Perancangan Perangkat Lunak**

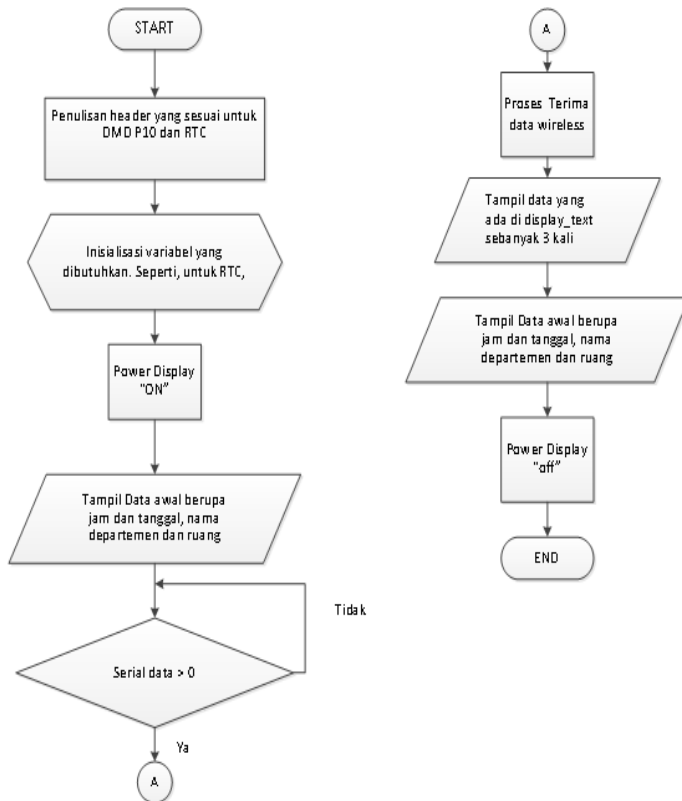
Dalam pembuatan alat ini, *software* yang dipakai untuk *display* menggunakan Arduino IDE 1.8.0. untuk program *Setting* Xbee menggunakan XCTU Version: 6.3.4 Build ID: 20161207-3 .

### 3.3.1 Flowchart

*Flowchart* program dari Tugas Akhir ini ini adalah meliputi seluruh sistem jalannya alat ini. Sistem yang dimaksud adalah sistem umum secara keseluruhan. **Gambar 3.15** pada gambar ini akan di mulai program untuk mengawali nya yaitu dengan meng-*upload* masing-masing program pada *slave*. Setelah program ter-*upload* pada masing-masing *slave* kita harus melakukan pengecekan koneksi antara Xbee *master* dan juga Xbee *slave* dengan melihat indikator dari *shield* Xbee. Apabila lampu d4 berkedip maka indikator menyatakan bahwa Xbee dalam keadaan terhubung. Namun apabila belum *connect* maka kita harus melakukan pemrograman ulang dan mungkin melakukan pengecekan parameter-parameternya.hal yang sangat penting adalah proses *upload master slave* . dimana *master* harus saling terhubung dengan *slave* . Setelah *master* terisi program kita harus mengecek port yang akan di gunakan untuk mengirim data serial ke Xbee *slave*. Setelah *master slave* mengirim perintah maka Xbee pada *slave* akan menjalankan perintah dan akan menghasilkan output yang sesuai dengan apa yang di inputkan pada *master slave*. Namun apabila tak ada data yang di kirim maka Xbee *slave* akan tetap berada pada keadaan *Standby*



**Gambar 3. 15** Flowchart Transfer Data Xbee



**Gambar 3. 16** Flowchart Program Penerimaan Slave

untuk flowchart **Gambar 3.16** Xbee slave adalah proses inisialisasi dari masing-masing library pada slave setelah proses selesai kita bisa menyalakan power supply dari slave. Pada keadaan awal slave akan menampilkan data berupa keadaan standby yang akan menampilkan jam, tanggal dan departemen teknik elektro otomasi. Dan apabila Xbee slave menerima data berupa string dari Xbee master maka slave akan melaksanakan programnya sebanyak perulangan yang di setting setelah

selesai eksekusi program maka *display* akan kembali kekeadaan awal. Apabila sudah menggunakan *display LED* maka jangan lupa untuk mematikan dengan mencabut catu daya.

### **3.3.2 Penyusunan Topology Jaringan Xbee**

Suatu jaringan XBee biasa disebut dengan PAN (*Personal Area Network*). Setiap jaringan ditetapkan dengan sebuah PAN *identifier* (PAN ID). XBee S2C menetapkan tiga macam perangkat *node* yang berbeda untuk membentuk jaringan, yaitu: *Coordinator*, *Router* dan *End device* sebagai media transmisinya.

#### **1. Coordinator**

*Coordinator* bertanggung jawab untuk membangun channel ber operasi dan PAN (*PersonalArea Network*) ID pada sebuah jaringan. *Coordinator* dapat membentuk suatu jaringandengan mengijinkan *router* dan *end device* untuk bergabung dalam jaringan tersebut. Setelah jaringan terbentuk, fungsi *Coordinator* seperti *router* (dapat berpartisipasi dalam *routing* paket dan menjadi sumber atau tujuan untuk paket data).

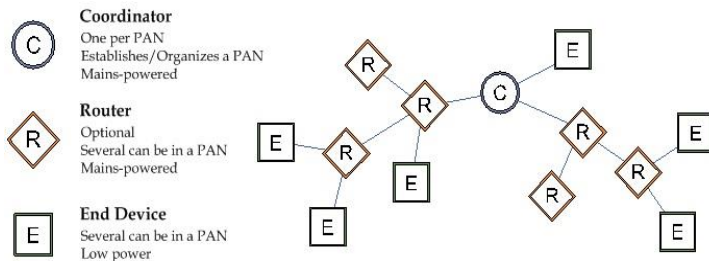
#### **2. Router**

Adalah sebuah *node* yang bertugas untuk menerima, mengirimkan dan merutekan data. Agar sebuah *router* dapat mengijinkan *router* lain dan *end device* untuk bergabung, maka *router* tersebut harus terlebih dahulu bergabung dengan jaringan PAN. Selain itu, *router* juga dapat berfungsi merutekan data antara PAN satu dengan yang lainnya. *Router* dapat berpartisipasi dalam *routing* paket dan menjadi sumber atau tujuan untuk paket data.

#### **3. End device**

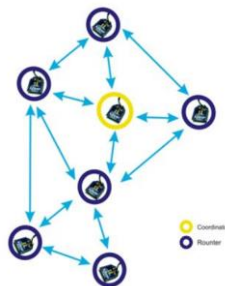
*End device* harus selalu berinteraksi atau terhubung dengan *Coordinator* atau *router* untuk dapat menerima dan mengirimkan data. *End device* juga dapat menjadi sumber atau tujuan untuk paket data tetapi tidak bisa untuk menentukan rute paket data.

Untuk menentukan bagaimana *topology* dalam proses ini bisa di lihat pada **Gambar 3.17** yang menjelaskan tentang *topology* jaringan Xbee



**Gambar 3. 17** Konsep Dasar Jaringan Xbee

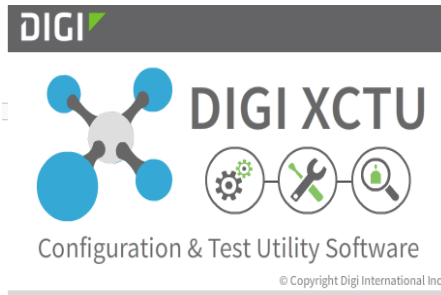
Setelah mengetahui dasar dari topology Xbee saya membuat sistem *topology* dengan proses *Mesh*. Pada sistem ini membutuhkan sedikitnya 1 *Coordinator* dan 2 buah *router* atau lebih untuk saling koneksi. Pada sistem ini pula sangat menguntungkan di karenakan dapat menambah daya jangkau dari Xbee dalam proses transmisi data. Pada proses ini data juga sangat mudah di karenakan *Coordinator* langsung mendapat keadaan langsungdari masing masing *router*. Untuk *topology* dari sistem *mesh* dan kami gunakan seperti **Gambar 3.18**.



**Gambar 3. 18** Konsep Jaringan *Mesh* Xbee

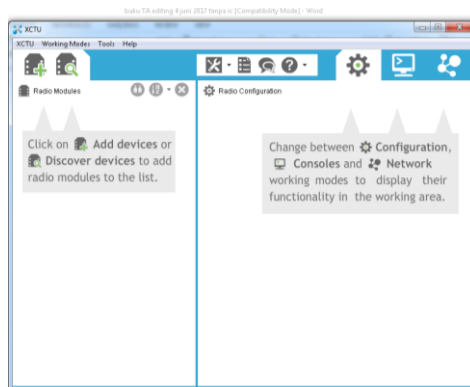
### 3.3.3 Setting Xbee dengan XCTU

Proses *setting* di lakukan pada program XCTU **Gambar 3.19** yang dimana program ini di peruntukkan untuk *setting* dari konfigurasi dari Xbee ini sendiri. Pada program ini juga dapat di lakukan proses simulasi untuk melakukan pengecekan apabila ingin menggunakan Xbee.



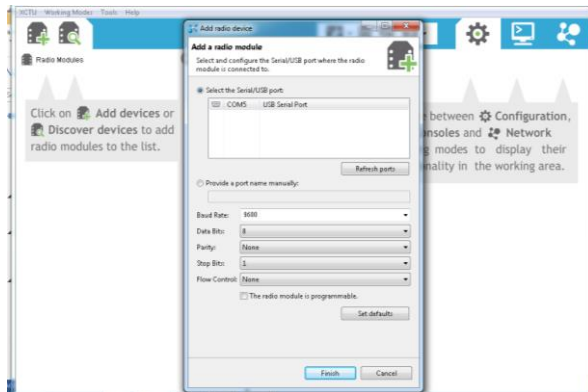
**Gambar 3. 19** Proses Pembukaan dari Program *XCTU*

Pada proses awal mula akan di tampilkan pada program DIGI **XCTU Gambar 3.20** yang di mana pada bagian ini menjelaskan tentang tes konfigurasi dan pengaturan-pengaturan yang menyangkut masalah Xbee. Apabila sudah melakukan proses pembukaan dari Xbee *programing* maka akan tampil halaman utama dari *XCTU* seperti gambar 3.26 pada gambar tersebut terdapat pengaturan untuk menambah modul-modul yang ingin di konfigurasi maupun ingin mengkonfigurasi modul yang tidak dalam sambungan laptop. Lalu ada pengaturan , *console* yang di gunakan untuk melakukan pengecekan dari modul modul yang sudah di setting dan ada juga suatu sistem kerja berdasarkan dari bentuk-bentuk *block* .



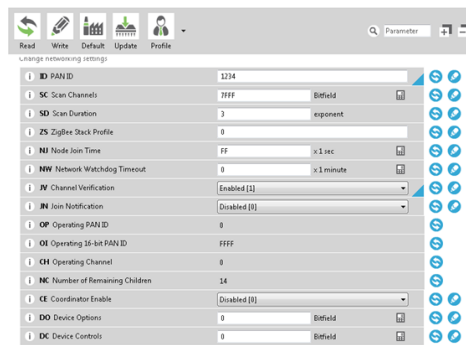
**Gambar 3. 20** Halaman Awal dari Program *XCTU*

Setelah program terbuka harus dilakukan *add module* untuk melakukan setting pada masing-masing Xbee yang akan di gunakan seperti **Gambar 3.21** pada gambar tersebut kita harus menyamakan *baud rate* dan pengaturan lain biarkan *default*.



**Gambar 3. 21** Halaman *Add Device*

Inti utama dari XCTU adalah di mana *Baud rate* kecepatan aliran data. pada dasarnya Xbee akan saling terhubung karena *Baud rate* nya sama maka memiliki kecepatan sama maka dari itu agar tidak saling tumpang tindih data maka di XCTU di lakukan konfigurasi. Untuk konfigurasi seperti **Gambar 3.22**.



**Gambar 3. 22** Gambar Parameter Xbee



Setelah mengetahui parameter parameter yang akan di gunakan dalam proses penyamaan konfigurasi antara Xbee *Coordinator* dan juga Xbee sebagai *router*. Sebelum mengisi konfigurasi kita terlebih dahulu harus menyamakan *firmware* dari masing-masing Xbee yang akan kita gunakan. Gunakan *Firmware* yang ter baru untuk mengantisipasi bila XCTU akan ada *updating*.

Parameter parameter yang akan di ganti yaitu:

PAN ID : berfungsi sebagai Kunci kerja dimana semua Xbee harus memiliki alamat yang sama agar dapat saling mengirim data.

DL (*Destination address low*) : titik kerja rendah pada Xbee.

DH (*Destination address High*) : titik kerja Tinggi pada Xbee.

NI (*Node Identifier*) : Nama spesifikasi Xbee.

CE(*Coordinator*) : untuk *Coordinator enable* dan untuk *router Disable*.

JV( *channel*) : Untuk *Coordinator Disable* dan *Router Enable*.

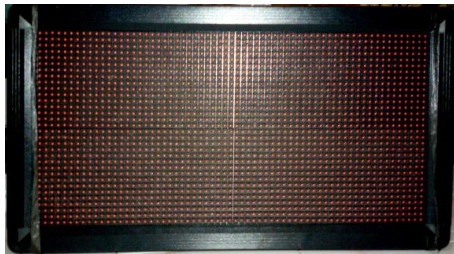
Setelah selesai bisa *check* hasil koneksi dari Xbee, Apakah Xbee yang dihubungkan sudah terhubung satu sama lain. Lalu juga *check* hasil pengiriman data yang telah di kirimkan .

-----Halaman ini sengaja dikosongkan-----

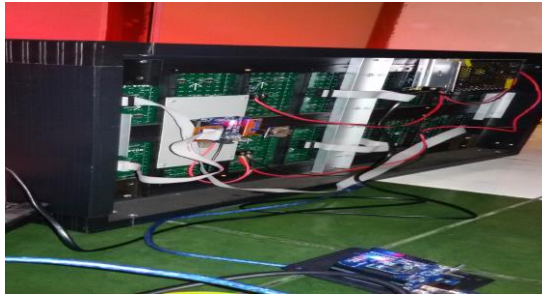
## BAB IV

### HASIL IMPLEMENTASI ALAT

Untuk mengetahui apakah tujuan-tujuan dari pembuatan alat ini telah sesuai dengan yang diharapkan atau tidak, maka dilakukan pengujian dan analisa terhadap alat yang telah dibuat **Gambar 4.1**. Alat yang telah dibuat yaitu sebuah *multi-display* yang di kontrol terpusat menggunakan *wireless*. Pada alat ini memiliki 1 buah *Coordinator (Master)* dan 2 buah *router (slave)*. Untuk tampilan awal dari masing-masing *slave* yaitu menampilkan jam dan waktu yang dimana menunjukkan informasi pokok dan keadaan *standby* dari *multi-display*. Selain jam dan juga tanggal tidak lupa ditampilkan pula tulisan yang di mana bertuliskan Departemen Teknik Elektro Otomasi yang bergantian terus menerus menyesuaikan dengan detik waktu. Setelah itu kami menampilkan ruang di mana *multi-display* di letakkan selama beberapa detik. Hal tersebut terjadi pada masing-masing *slave* secara terus menerus hingga masing-masing *slave* menerima sinyal data yang dikirimkan dari Xbee dengan alamat yang sama. Untuk *Coordinator* ditancapkan pada *PC master* lalu buka *software* Arduino setelah itu lihat pada bagian port yang terhubung dengan *master slave*. Tulis pengumuman yang akan di sampaikan pada *multi-display*. Untuk proses pengiriman harus menunggu program *standby* selesai di eksekusi setelah itu maka informasi pengumuman akan di distribusikan pada masing-masing *slave* yang akan di gunakan.



**Gambar 4. 1** Hasil Slave Multi-Display



**Gambar 4. 2** Hasil Modul Komunikasi Multi-Display

Setelah informasi di terima oleh Xbee maka informasi akan mengikuti perintah program yang ada dalam *slave*. Mulai dari berapa kali proses pengumuman akan di lakukan hingga besar huruf akan di sesuaikan dengan pengaturan dari *slave*. Setelah program eksekusi pada *slave* berakhir maka akan di kembalikan pada tampilan *standby*. Dari **Gambar 4.2** adalah hasil pembuatan dan pengkoneksian antara modul komunikasi yang akan menghubungkan *PC master* dan *slave multi-display*.

#### **4.1 Pengujian Multi-Display**

Pada Subbab ini akan dijelaskan mengenai hasil pengujian dari *multi-display* yang dibuat. Untuk mengetahui lebih lanjut apakah sudah sesuai dengan teori dan juga komponen komponen penyusun sudah berjalan semestinya dan tanpa adanya kendala. Pengujian pengiriman data dari hasil perancangan alat seperti yang telah di bahas pada **BAB 3**, dan juga proses data *transfer range* pada proses pengiriman data. Tujuan pengujian alat ini adalah untuk mengetahui ketepatan dan ketelitian dari alat yang dibuat, sehingga dengan melakukan pengujian-pengujian secara bertahap pada modul komunikasi *multi-display* hingga diketahui kekurangan-kekurangan yang mungkin bisa untuk lebih diperbaiki lagi kedepannya. Dimana dari dua *slave multi-display* yang dibuat, penulis hanya mengambil data pada satu salah satu *slave* saja, dikarenakan kesamaan perancangan perancangan dan pemrogramannya.

##### **4.1.1 Pengujian Data yang Dikirim**

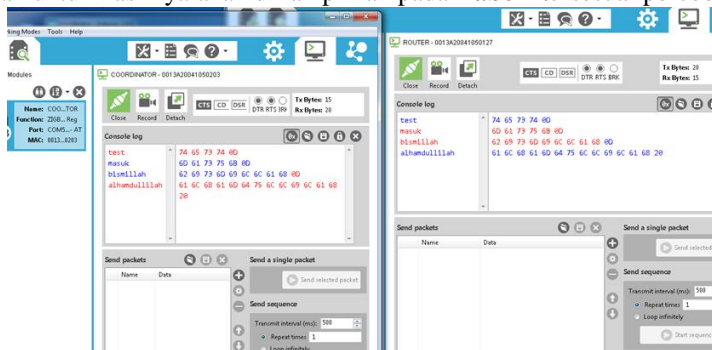
Untuk pengiriman data melalui Xbee bisa di cek menggunakan XCTU. Pada program XCTU bisa menampilkan data yang dikirim dan

data yang akan di terima oleh Xbee penerima. Pada proses konfigurasi Xbee kita harus terlebih dahulu men-*setting* konfigurasi Xbee seperti dengan menerapkan jaringan *Mesh* seperti tabel **Tabel 4.4**.

**Tabel 4. 4 Konfigurasi Xbee pada XCTU**

NO	Konfigurasi	<i>Coordinator</i>	<i>Router</i>
1.	PAN ID	1234	1234
2.	DL (Destination address low)	FFFF	FFFF
3.	DH (Destination address High)	0	0
4.	NI ( <i>Node Identifier</i> )	<i>COORDINATOR</i>	<i>ROUTER</i>
5.	CE( <i>Coordinator</i> )	DISABLE	ENABLE
6.	JV( channel)	ENABLE	DISABLE

Setelah proses *setting* dari Xbee pada Arduino telah selesai kita akan masuk ke tahap pengecekan data yang akan kita transmisikan **Gambar 4.3**. Untuk proses ini di perlukan setidaknya 2 buah Xbee yang saling terhubung dengan PC dan di jalankan pada program XCTU untuk prosesnya kita masuk ke dalam *console* untuk mengetahui data yang akan di kirim menggunakan Xbee apa sudah sesuai dan sama apa belum. Dan untuk hasilnya akan di lampirkan pada **Tabel 4.5** sesuai percobaan.



**Gambar 4. 3** Pengujian dengan Xbee

**Tabel 4. 5 Data Pengiriman**

Daftar Pengiriman Data menggunakan XCTU			
<i>Coordinator</i>	Code	<i>Router</i>	Code
TEST	74 65 73 74 OD	TEST	74 65 73 74 OD

MASUK	6D 61 73 75 6D	MASUK	6D 61 73 75 6D
ALHAMDULILLAH	61 6C 68 61 6D 64 75 6C 69 6C 6C 61 68	ALHAMDULIL LAH	61 6C 68 61 6D 64 75 6C 69 6C 6C 61 68
DONE	64 6F 6E 65	DONE	64 6F 6E 65

Untuk proses pengiriman data ini bisa berlangsung banyak data dalam sekali kirim namun untuk proses penerimaan dari *slave multi-display* hanya terbatas. range komunikasi bisa mencapai 2 kali dari transmisi biasa . pada inisialisasi awal yang digunakan untuk menyimpan karakter. Tipe data *byte* memiliki 8 *bits data*, yang memiliki *range* 0 – 255. Tetapi ketika penulis mengganti tipe data menjadi yang lain, seperti *word*, tetap saja menghasilkan karakter maksimal sebanyak 63 karakter. berarti 1 karakter pada *multi-display* mewakili sebanyak 4 satuan *range* pada tipe data *byte*.

#### 4.1.2 Pengujian *Range Komunikasi Multi-Display*

Pada proses ini kami melakukan pengukuran guna mengetahui proses pengiriman data pada setiap setiap *slave*. Pada pengujian ini kami lakukan di jalan setapak antara fakultas Vokasi dan Fakultas Sipil. Untuk proses pengukuran kami melakukan dengan proses pengambilan data per 10 meter jarak. Untuk proses pengambilan data harus berada di luar ruang yang tidak tertutup (*lineoffset*) dan hasil di tunjukkan pada **Tabel 4.6**.

**Tabel 4. 6 Jarak Pengukuran Diluar Ruang**

NO	Jarak	Status
1.	0 meter	Terkirim
2.	10 meter	Terkirim
3.	20 meter	Terkirim
4.	30 meter	Terkirim
5.	40 meter	Terkirim
6.	50 meter	Terkirim
7.	60 meter	Terkirim
8.	70 meter	Terkirim
9.	80 meter	Terkirim

10.	100 meter	Terkirim
11.	110 meter	Tidak Terkirim
12.	120 meter	Tidak Terkirim

Berikut adalah data hasil pengukuran jarak pengiriman dari *multi-display* dari *server PC* ke masing-masing *Slave*. Pengiriman ini bersifat sementara setelah proses dari *slave* selesai di eksekusi maka akan kembali ke keadaan *stanby* dari masing-masing *slave* dengan menampilkan.

Untuk data ke 2 yaitu pengukuran dalam ruang , pengukuran dalam ruang di lakukan di gedung Department Teknik Elektro Otomasi ITS hasil pada **Tabel 4.7**.

**Tabel 4. 7 Jarak Pengukuran Didalam Ruang**

NO	Jarak	Status
1.	0 meter	Terkirim
2.	2 meter	Terkirim
3.	5 meter	Terkirim
4.	10 meter	Terkirim
5.	15 meter	Terkirim
6.	20 meter	Terkirim
7.	25 meter	Tidak Terkirim

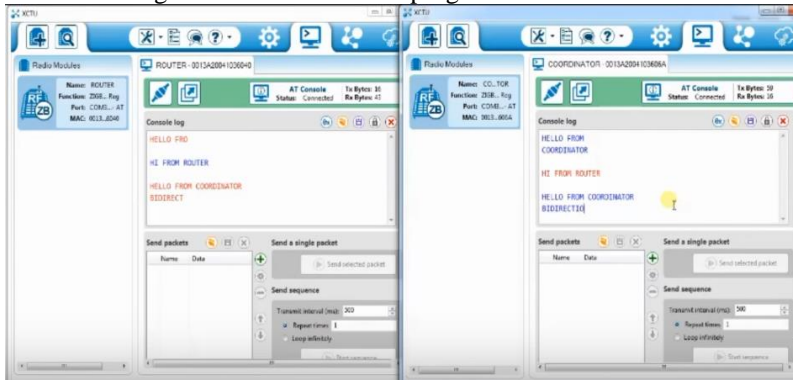
Dari pengukuran dalam ruang di dapatkan data sejauh 20 meter data masih terkirim namun terkadang data juga mengalami *loss* (hilang) di karenakan Xbee berada di luar jangkauan inilah hasil dari pengujian kami tanpa adanya rekayasa

## 4.2 Pengujian Keseluruhan Sistem

Pada subbab ini, akan dipaparkan mengenai pengujian dari keseluruhan sistem yang digunakan dalam Tugas Akhir ini. Pengujian ini didasarkan pada diagram blok keseluruhan sistem yang terdapat pada **BAB III** di atas yang menjelaskan mengenai hubungan perangkat *Master* dengan perangkat *Slave* melalui komunikasi *wireless* XBee yang ada pada kedua perangkat tersebut. Hasil pengujian ini membahas keunggulan dari *multi-display* dikontrol terpusat menggunakan *wireless* secara keseluruhan yaitu:

#### 4.2.1 Hasil Pengiriman Data Komunikasi

Hasil dari Tugas Akhir ini dapat diketahui bahwa pengiriman data dari *PC master* ke *slave* sudah sesuai dengan yang dikirim dari *PC master*. Untuk pengiriman data komunikasi dengan Xbee memiliki presentasi kesesuaian pengiriman hingga 100% dari beberapa hasil pengujian yang terdapat pada **Tabel 4.5**. data komunikasi yang dikirim berupa text. Untuk pengiriman kesesuaian data dapat dipantau dari program XCTU. Pada Xbee *master slave* akan mengirim data informasi ke Xbee *slave* untuk dieksekusi. Setelah eksekusi selesai Xbee *master* akan menerima *feedback* berupa data yang diproses dari *slave*. Di sini kita akan mengetahui kesesuaian dari program awal instalasi.



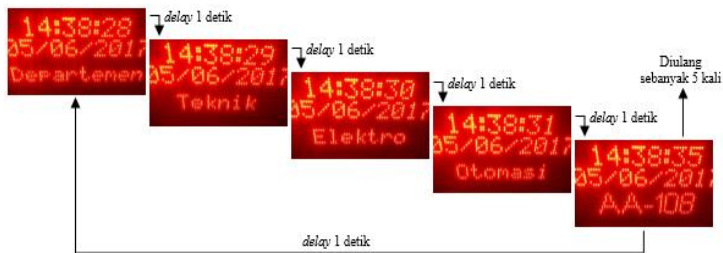
**Gambar 4. 4** Pengecekan dengan XCTU

pada program XCTU **Gambar 4.4** dapat mengatur proses instalasi dari Xbee untuk menentukan titik kerja dan *topology*-nya. Dalam kegiatan ini kami menggunakan *topology mesh*. Yang akan menambah daya jangkau dari Xbee ke masing-masing *slave* dengan menjadikan *slave* sebagai *repeater* menuju ke *slave* yang terjauh .

#### 4.2.2 Hasil Tampilan Multi-Display

Hasil dari alat kami menampilkan tampilan awal seperti **Gambar 4.5** dimana pada tampilan *multi-display* menampilkan jam, tanggal dan tempat di pasang nya *multi-display*. Pada alat yang kami buat dapat menampilkan waktu yang sesuai dengan *master PC* dengan tepat. Dari waktu *multi-display* ini memiliki *error time* sangat kecil sehingga jam yang di tampilkan akan sesuai terus dengan *PC master*.





**Gambar 4. 5** Tampilan Multi-Display

pada gambar di atas menunjukkan tampilan yang *default* dari *multi-display*. Tampilan ini dapat di rubah pada saat kita melakukan pemrograman pada *slave*. Proses pemrograman ini tidak akan memperngaruhi kondisi dari waktu *multi-display*. Namun apabila waktu *ter-reset* maka harus di lakukan instalisasi ulang pada *mult-display*. Dari nyala tampilan *multi-display* dapat di lihat hingga kejauhan 10 meter. Untuk pengumuman dari *master slave* dapat di lihat setelah program *default* tereksekusi. Hasil dari pengiriman *master slave* dapat di lihat pada **Gambar 4.6**



**Gambar 4. 6** Tampilan Pengumuman Multi-Display

#### 4.2.3 Hasil Display Pengiriman dan Waktu Transmisi

Dalam hal banyaknya karakter yang dapat dikirimkan, baik melalui sambungan antara perangkat *slave multi-display* yang langsung terhubung dengan PC atau Laptop melalui kabel USB maupun melalui komunikasi *wireless* XBee yang ada di perangkat *master* dan *slave*

jumlah karakter yang dapat dikirimkan maksimal hanya 63 karakter.

**Gambar 4. 7** memperlihatkan perangkat *master* dari Tugas Akhir ini. Pengujian yang dilakukan adalah menuliskan karakter sebanyak 70 karakter, namun yang dapat ditampilkan hanya 63 karakter (karakter yang dimasukkan sama dengan pengujian pada pembahasan sebelumnya). Dapat di lihat pada **Lampiran A.5**.



**Gambar 4. 7** Koneksi PC dan Modul Komunikasi

Kisaran waktu dalam menampilkannya jika menggunakan komunikasi *wireless* XBee antara dua perangkat tersebut terdapat perbedaan waktu, namun tidak terlalu jauh berbeda. Jika perangkat *Slave* langsung terhubung dengan PC atau Laptop berkisar antara 0,5 – 1,8 detik, maka dengan komunikasi *wireless* XBee memerlukan waktu 1 – 2,2 detik

#### 4.2.4 Hasil Jarak yang Dapat Dijangkau oleh XBee S2C

Dalam pengujian jarak yang dapat dijangkau oleh XBee S2C, dilakukan 2 pengujian untuk yang tanpa halangan (LOS/*Line of Sight*) dan dengan halangan (dipisahkan ruang).

Pengujian tanpa halangan (LOS), dilakukan secara bertahap dengan pengiriman data, yaitu dari jarak 0 – 120 meter. Setelah data dikirimkan, dalam jangkauan jarak yang berkisar antara 0 – 100 meter, perangkat *slave multi-display* masih dapat menampilkan data yang dikirimkan, sedangkan selebihnya, data tidak ditampilkan oleh perangkat *slave* tersebut. Sehingga, jarak maksimum yang dapat dijangkau secara *wireless* tanpa halangan adalah 100 meter seperti **Tabel 4.6**.

Pengujian dengan halangan (dipisahkan ruang), juga dilakukan secara bertahap dengan pengiriman data, yaitu dari jarak 0 – 50 meter. Setelah data dikirimkan, dalam jangkauan jarak yang berkisar antara 0 – 20 meter, perangkat *slave multi-display* masih dapat menampilkan data

yang dikirimkan, sedangkan selebihnya, data tidak ditampilkan oleh perangkat *slave* tersebut. Sehingga, jarak maksimum yang dapat dijangkau secara *wireless* dengan halangan adalah 20 meter seperti **Tabel 4.7**. Dari **Gambar 4. 8** menunjukkan hasil tampilan *multi-display* yang menerima data di dalam ruang yang berbeda.



**Gambar 4. 8** Tampilan *Multi-Display* di Dalam Ruang

-----Halaman ini sengaja dikosongkan-----

## **BAB V**

### **PENUTUP**

Setelah melakukan perencanaan, perancangan, dan pengujian alat. maka dapat mengambil kesimpulan dan memberikan saran demi penyempurnaan Tugas Akhir ini.

#### **5.1 Kesimpulan**

Hasil dari perancangan serta implementasi alat dari *Multi-display* yang Dikontrol Secara Terpusat Menggunakan *Wireless* dapat diambil kesimpulan sebagai berikut:

1. Proses Pengiriman Data dalam ruang terbuka setelah kami lakukan proses uji coba mencapai 100 meter
2. Proses Pengiriman Data dalam gedung setelah kami lakukan proses uji coba mencapai 20 meter
3. Untuk Xbee harus memiliki Baud Rate yang sama untuk melakukan pengiriman data pada masing-masing Xbee yang akan di gunakan.
4. Dalam penelitian Kami menggunakan sistem jaringan *Mesh* untuk menambah daya jangkau range komunikasi menjadi 2 kali.
5. Jumlah karakter yang dapat diterima *Slave* memiliki batas maksimal yang harus ditaati, yaitu sebanyak 63 karakter.
6. pada proses pengiriman data dapat terpantau pada *master slave* dan akan di lakukan perulangan sebanyak program masing-masing *slave*.
7. Xbee memiliki kriteria untuk mengirim data dalam Baud rate yang sama dan PAN ID yang sama untuk menghindari data bertubrukan.

#### **5.2 Saran**

Untuk pengembangan dan penyempurnaan pembuatan *Multi-display* yang di kontrol terpusat, maka diberikan beberapa saran sebagai berikut:

1. Pada pengembangan selanjutnya di harapkan dapat mengatur tanggal , waktu dan tempat dengan sistem *Wireless*. Tanpa perlu instalisasi masing-masing *slave*
2. Dalam penelitian selanjutnya, diharapkan dapat mengirimkan data lebih dari 63 karakter. Selain itu, juga perlu ditambahkan variasi

yang menunjang penampilan dan performa dari *multi-display* ini, baik dari tampilan maupun pemrogramannya.

## DAFTAR PUSTAKA

- [1] Ahamed, Dr.S.S.Riaz. 2005. "The Role of Zigbee Technology in Future Data Communication System".Sathak Institute of Technology.
- [2] Azam Muzakhim, "Telemetri dan Telekontrol Antar Mikrokontroler menggunakan Xbee-pro wireless", Jurnal ELTEK, 2011, Volume 09 Nomor 02, Oktober 2011, ISSN 1693-4024.
- [3] B. Azzedinne. 2005. "Algorithms and Protocols for Wireless Sensor Networks". John Wiley and Sons.
- [4] Chandani Mauriya, Dr. M. B. Mali, "*Wireless* LED Matrix Display System, 2016 International Journal of Innovative Research in Computer and Communication Engineering, Pune, 2016, Vol. 4, Issue 6, 193\_*Wireless*.pdf
- [5] Faludi Robert.2011 Building *Wireless* Sensor Networks - With Zigbee, Xbee, Arduino, And Processing
- [6] Firdaus. (2014). *Wireless* Sensor Network Teori dan Aplikasi. Yogyakarta: Graha Ilmu
- [7] Gaurav S. Yawale,2016." Design of an Intelligent System for Highway using Wireless Sensor Network" Department of EXTC. Amravati.
- [8] Hamdani, Fadil. "Pemodelan dan Simulasi Jaringan Sensor Nirkabel Micaz Mote Berdasarkan Standar IEEE 802.15.4 ". 2010.
- [9] Inc, D. I. 2008.X-CTU Configuration & Test Utility *Software*.Minnetonka: DigiInternational Inc.
- [10] Khamdan Amin Bisyr, "Rancang BangunKomunikasi Data *Wireless* MikrokontrolerMenggunakan Modul Xbee Zigbee

(IEEE802.15.4)”, Departemen Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Institut Pertanian Bogor, Bogor, 2012.

- [11] M. A. B. Sarijari, R. A. Rashid, M. R. A. Rahim, and N. H. Mahalin, "*Wireless* Home Security and Automation System Utilizing ZigBee based Multi-hop Communication," in Telecommunication Technologies 2008 and 2008 2nd Malaysia Conference on Photonics. NCTT-MCP 2008.6th National Conference on, 2008, pp. 242-245.
- [12] M. A. Kader, M. M. Rahman, S. M. I. B. Haider and M. Islam, "LED matrix based digital learning display for children with *wireless* control," 2014 17th International Conference on Computer and Information Technology (ICCIT), Dhaka, 2014, pp. 397-400. doi: 10.1109/ICCITech.2014.7073134
- [13] Rajeev Piyare. 2013. "Performance Analysis of XBee ZB Module Based Wireless Sensor Networks". Mokpo National University
- [14] Robert Faludi. 2011 Building *Wireless* Sensor Networks - With Zigbee, Xbee, Arduino, And Processing - - O'Reilly
- [15] Said, Pradana. 2010 Performance-Analysis-of-Xbee-ZB-Module-Based-*Wireless*-Sensor-Networks. ITB
- [16] Sayekti, Ilham. 2013." Bel Pemanggil Perawat Berbasis Wireless Menggunakan Xbee". Jurusan Teknik Elektro Politeknik Negeri Semarang
- [17] Sonal A. Mahajan. 2016." Monitoring Parameter of Distribution Transformer by using XBEE Technology". DES's College of engineering and technology
- [18] Sugiarto, B. (2010). Perancangan Sistem Pengendalian Suhu pada Gedung Bertingkat dengan Teknologi *Wireless* Sensor Network. Jurnal Ilmiah Teknik Mesin Cakra M , 62-68.



- [19] Susana, Ratna., dkk. (2015). Implementasi *Wireless* Sensor Network Prototype Sebagai Fire Detector menggunakan Arduino Uno. *Jurnal Elektro Telekomunikasi Terapan* .
- [20] Wibowo, S. E., 2015. Pengetahuan Dasar RTC DS1307. Gramedia :Jakarta
- [21] Winardi. 2010. “Mengenal Teknologi ZigBee Sebagai Standart Pengiriman Data Secara Wireless”. Binus University.

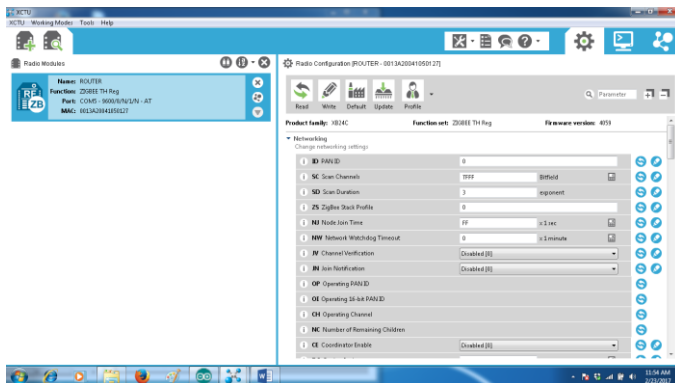
-----Halaman ini sengaja dikosongkan-----

## LAMPIRAN

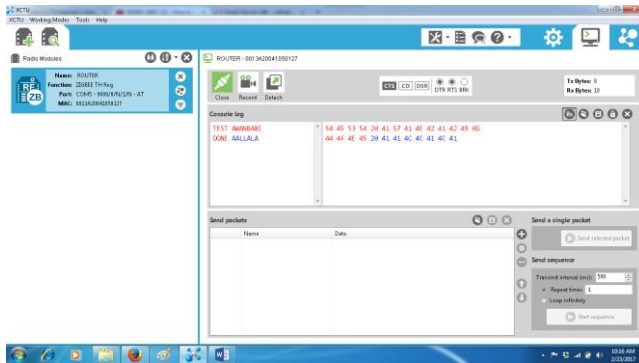
### A.1. Dokumentasi Pada Masa Percobaan Koneksi Xbee



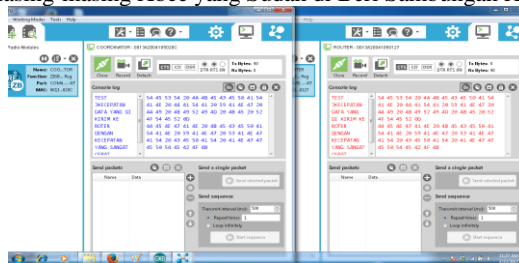
Gambar 1. Pengecekan Xbee dengan XCTU



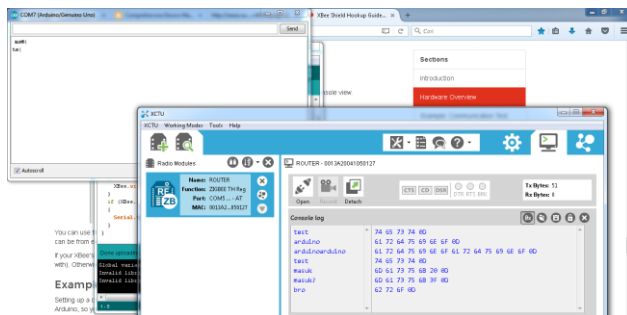
Gambar 2. Proses Pemberian Konfigurasi Xbee



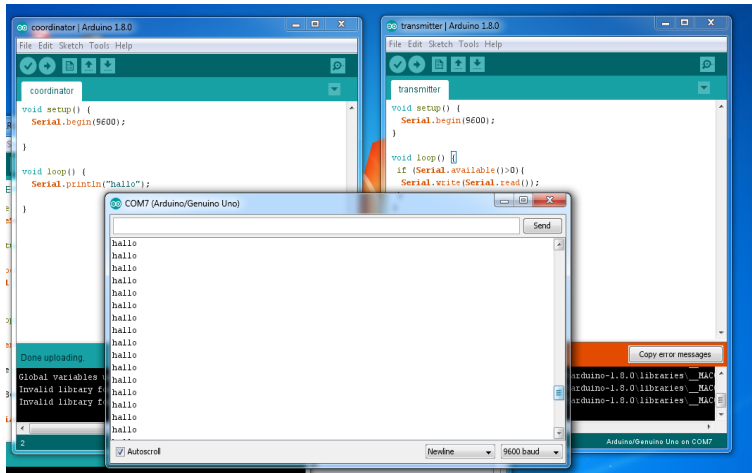
**Gambar 3.** Pengiriman Data Melalui 2 Buah PC yang Dikonekkan pada Masing-masing Xbee yang Sudah di Beri Sambungan Xbee



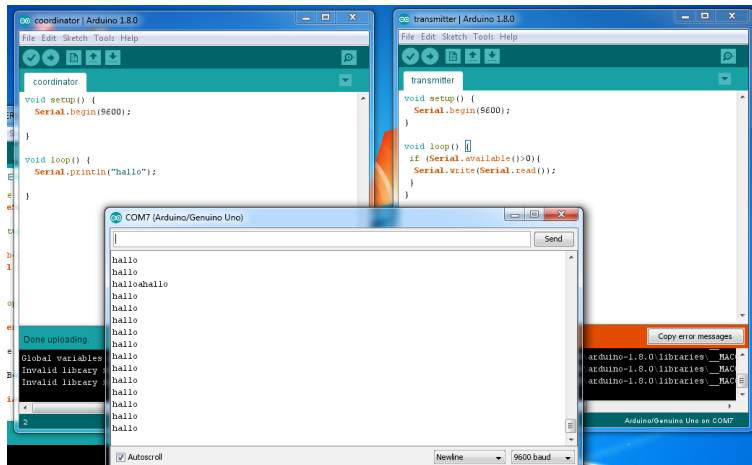
**Gambar 4.** Percobaan Menggunakan 2 modul Xbee



**Gambar 5.** Proses Penghubungan Xbee Arduino Uno dengan Xbee dalam Program XCTU



**Gambar 6** Proses Test dengan Proses tanpa Halangan yang Dicoba dengan Jarak yang Dekat Dengan Laptop Berjejer

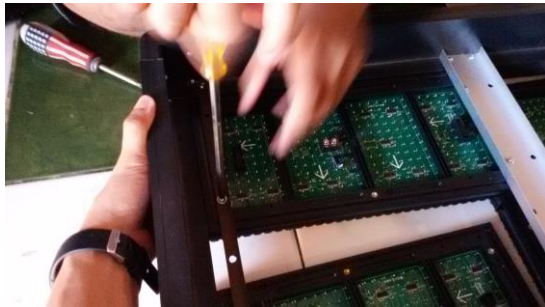


**Gambar 7** Proses Hasil Test dengan Jarak Sekitar 40 m

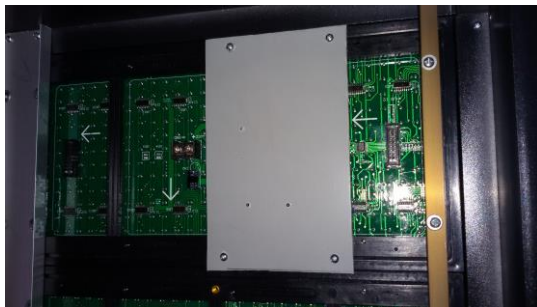
## A.2. Dokumentasi Proses Perakitan *Multi-Display*



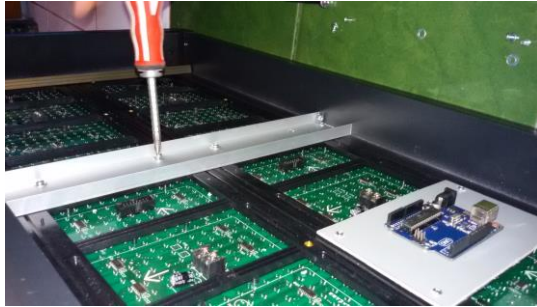
**Gambar 8.** Bingkai *Multi-Display*



**Gambar 9.** Pemasangan Modul DMD P10 pada Bingkai



**Gambar 10.** Pemasangan Papan yang Digunakan untuk Menempatkan Kontroler Arduino



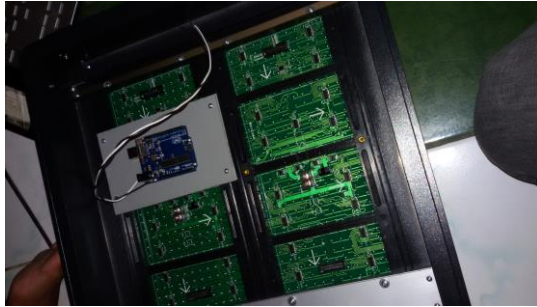
**Gambar 11.** Pemasangan Tahap Akhir dari Modul DMD P10



**Gambar 12.** Hasil yang Didapatkan Setelah Pemasangan



**Gambar 13.** Pemasangan *Power Supply Switching*



**Gambar 14.** Pemasangan Kabel *Power* pada Kontroler dan Modul DMD dari *Power Supply*



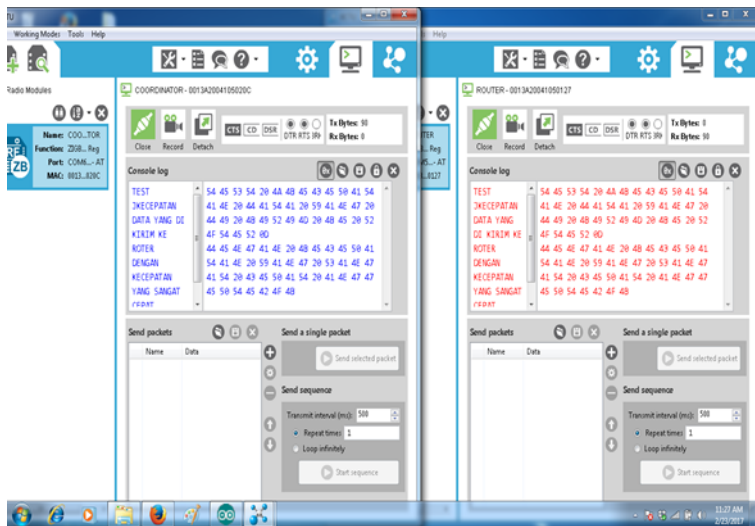
**Gambar 15.** Penyambungan Antara Modul DMD 1 dengan yang lain Secara Seri Menggunakan Kabel *Ribbon*



**Gambar 16.** Hasil Tampak Depan *Multi-display*



### A.3. Pengujian Data Xbee



Gambar 17. Pengiriman Data XCTU pada Xbee

### A.4. Hasil Pengujian *RangeKomunikasi*

Tabel 1 Jarak Pengukuran Luar Ruang

NO	Jarak	Status
1.	0 meter	Terkirim
2.	10 meter	Terkirim
3.	20 meter	Terkirim
4.	30 meter	Terkirim
5.	40 meter	Terkirim
6.	50 meter	Terkirim
7.	60 meter	Terkirim
8.	70 meter	Terkirim
9.	80 meter	Terkirim
10.	100 meter	Terkirim
11.	110 meter	Tidak Terkirim
12.	120 meter	Tidak Terkirim

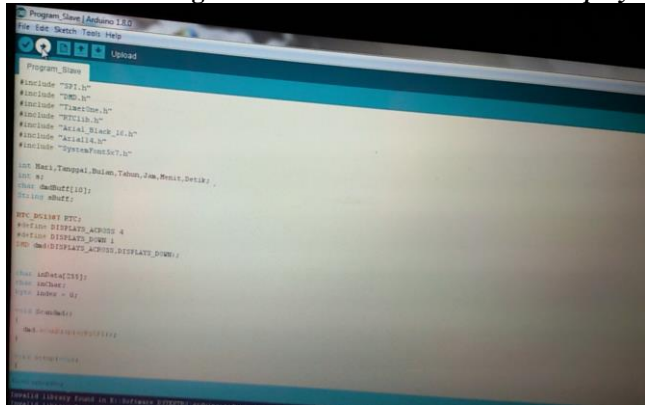


**Gambar 18.** Pengiriman Data *Multi-Display*

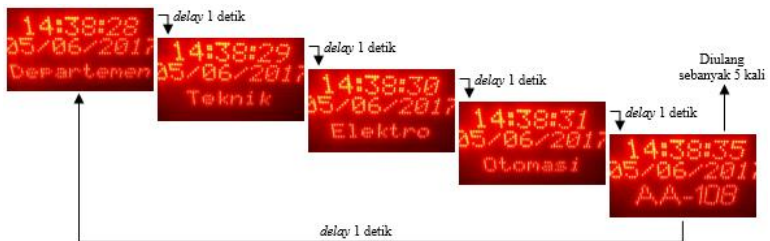
**Tabel 2** Jarak Pengukuran Dalam Ruang

NO	Jarak	Status
1.	0 meter	Terkirim
2.	2 meter	Terkirim
3.	5 meter	Terkirim
4.	10 meter	Terkirim
5.	15 meter	Terkirim
6.	20 meter	Terkirim
7.	25 meter	Tidak Terkirim

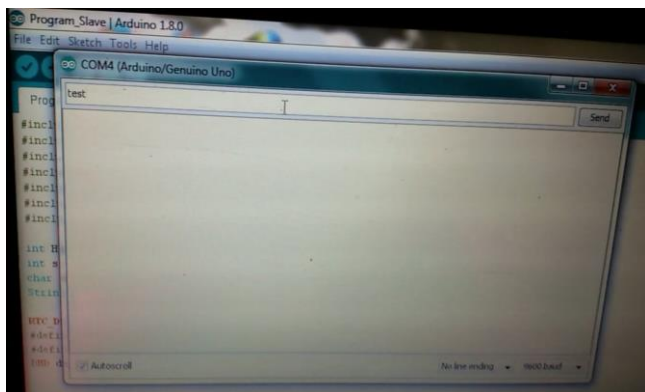
### **A.5 Dokumentasi Pengiriman Data dari PC ke *Multi-Display***



**Gambar 19.** *Compiling dan Uploading Program*



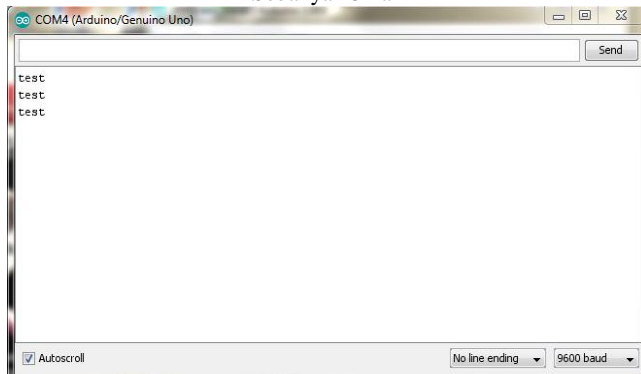
**Gambar 20.** Ilustrasi Tampilan *Standby* pada *Multi-display*



**Gambar 21.** Pengiriman Data Pengumuman Melalui *Serial Monitor*



**Gambar 22.** Pengumuman Ditampilkan pada *Multi-display* Sebanyak 3 kali



**Gambar 23.** Monitoring Data yang Dikirimkan dari *Serial Monitor* dan Ditampilkan oleh *Multi-display*

## A.6. Listing Program

```
#include "SPI.h"
#include "DMD.h"
#include "TimerOne.h"
#include "RTClib.h"
#include "Arial_Black_16.h"
#include "Arial14.h"
#include "SystemFont5x7.h"

int Hari,Tanggal,Bulan,Tahun,Jam,Menit,Detik;
int s;
char dmdBuff[10];
String sBuff;

RTC_DS1307 RTC;
#define DISPLAYS_ACROSS 4
#define DISPLAYS_DOWN 1
DMD dmd(DISPLAYS_ACROSS,DISPLAYS_DOWN);

#define max_char 100
char inData[max_char];
char inChar;
byte index = 0;

void Scandmd()
{
    dmd.scanDisplayBySPI();
}

void setup(void)
{
    RTC.begin();
    RTC.adjust(DateTime(F(__DATE__), F(__TIME__)));
    Serial.begin(9600);
    Timer1.initialize( 5000 );
    Timer1.attachInterrupt( Scandmd );
    dmd.clearScreen( true );
```

```

}

void(* ku_reset) (void) = 0;

void loop(void)
{
  while(Serial.available() == 0)
  {
    if (index > 0)
    {
      for(int i=0;i<3;i++)
      {
        running_text();
      }
      ku_reset();
    }else{
      dept_name();
      break;
    }
  }
  while(Serial.available() > 0)
  {
    dmd.clearScreen(true);
    if(index < max_char-1)
    {
      inChar = Serial.read();
      inData[index] = inChar;
      index++;
      inData[index] = 0;
      break;
    }
  }
}

void running_text()
{
  Serial.println(inData);
  dmd.clearScreen(true);

```

```

dmd.selectFont(Arial_14);
dmd.drawMarquee(inData,max_char,(32*4)-1,1);
long start=millis();
long timer=start;
boolean ret=false;
while(!ret)
{
    if ((timer+50) < millis())
    {
        ret=dmd.stepMarquee(-1,0);
        timer=millis();
    }
}
}

void TampilWaktu()
{
    DateTime now = RTC.now();
    Tanggal = now.day();
    Bulan = now.month();
    Tahun = now.year();
    Jam = now.hour();
    Menit = now.minute();
    Detik = now.second();
    dmd.selectFont(System5x7);
    dmd.drawFilledBox( 23, 1, 24, 2, GRAPHICS_OR );
    dmd.drawFilledBox( 23, 4, 24, 5, GRAPHICS_OR );
    dmd.drawFilledBox( 38, 1, 39, 2, GRAPHICS_OR );
    dmd.drawFilledBox( 38, 4, 39, 5, GRAPHICS_OR );
    dmd.drawString(15, 8, "/",1, GRAPHICS_OR);
    dmd.drawString(33, 8, "/",1, GRAPHICS_OR);

    // Display JAM
    sBuff = Convert(now.hour());
    s = sBuff.length()+1;
    dmdBuff[s];
    sBuff.toCharArray(dmdBuff,s);

```

```
dmd.selectFont(System5x7);  
dmd.drawString(11,0,dmdBuff,s,0);
```

```
// Display Menit  
sBuff = Convert(now.minute());  
s = sBuff.length()+1;  
dmdBuff[s];  
sBuff.toCharArray(dmdBuff,s);  
dmd.selectFont(System5x7);  
dmd.drawString(26,0,dmdBuff,s,0);
```

```
// Display Detik  
sBuff = Convert(now.second());  
s = sBuff.length()+1;  
dmdBuff[s];  
sBuff.toCharArray(dmdBuff,s);  
dmd.selectFont(System5x7);  
dmd.drawString(41,0,dmdBuff,s,0);
```

```
//Display Tanggal  
sBuff = Convert(now.day());  
s = sBuff.length()+1;  
dmdBuff[s];  
sBuff.toCharArray(dmdBuff,s);  
dmd.selectFont(System5x7);  
dmd.drawString(3,8,dmdBuff,s,0);
```

```
//Display Bulan  
sBuff = Convert(now.month());  
s = sBuff.length()+1;  
dmdBuff[s];  
sBuff.toCharArray(dmdBuff,s);  
dmd.selectFont(System5x7);  
dmd.drawString(21,8,dmdBuff,s,0);
```

```
//Display Tahun  
sBuff = Convert(now.year());
```



```

s = sBuff.length()+1;
dmdBuff[s];
sBuff.toCharArray(dmdBuff,s);
dmd.selectFont(System5x7);
dmd.drawString(39,8,dmdBuff,s,0);
}

void dept_name()
{
    dmd.clearScreen(true);
    TampilWaktu();
    dmd.selectFont(System5x7);
    dmd.drawString(67,4,"Departemen",10,GRAPHICS_OR);
    delay(1000);
    dmd.clearScreen(true);
    TampilWaktu();
    dmd.selectFont(System5x7);
    dmd.drawString(78,4,"Teknik",6,GRAPHICS_OR);
    delay(1000);
    dmd.clearScreen(true);
    TampilWaktu();
    dmd.selectFont(System5x7);
    dmd.drawString(76,4,"Elektro",7,GRAPHICS_OR);
    delay(1000);
    dmd.clearScreen(true);
    TampilWaktu();
    dmd.selectFont(System5x7);
    dmd.drawString(76,4,"Otomasi",7,GRAPHICS_OR);
    delay(1000);
    for(int b=0;b<5;b++)
    {
        dmd.clearScreen(true);
        TampilWaktu();
        dmd.selectFont(Arial_14);
        dmd.drawString(75,1,"AA-108",6,GRAPHICS_OR);
        delay(1000);
    }
}

```

```
}
```

```
String Convert(int Num){  
    if(Num < 10){  
        return "0"+String(Num);  
    }  
    else{  
        return String(Num);  
    }  
}
```

## A.7. Datasheet

<i>Specification</i>	<b>Xbee</b>	<b>Xbee-PRO (S2C)</b>
<b>Indoor/Urban Range</b>	up to 133 ft. (40 m)	up to 200 ft (60 m)
<b>Outdoor RF line-of-sight Range</b>	up to 400 ft. (120 m)	up to 5000 ft (1500 m)
<b>RF Data Rate</b>	250,000 bps	250,000 bps
<b>Data Throughput</b>	up to 35000 bps	up to 35000 bps
<b>Serial Interface Data Rate (software)</b>	1200 bps - 1 Mbps	1200 bps - 1 Mbps
<b>Receiver Sensitivity</b>	-96 dBm, boost mode enabled -95 dBm, boost mode disabled	-102 dBm
<b>Supply Voltage</b>	2.1 - 3.6 V	3.0 - 3.4 V
<b>Operating Current (Transmit, max output power)</b>	40mA (@ 3.3 V, boost mode enabled) 35mA (@ 3.3 V, boost mode disabled)	295mA (@3.3 V) 170mA (@3.3 V) international variant
<b>Operating Current (Receive)</b>	40mA (@ 3.3 V, boost mode enabled) 38mA (@ 3.3 V, boost mode disabled)	45 mA (@3.3 V)
<b>Idle Current (Receiver)</b>	15mA	15mA
<b>Power-down Current</b>	< 1 uA @ 25 C	3.5 uA
<b>Operating Frequency</b>	ISM 2.4 GHz	ISM 2.4 GHz
<b>Dimensions</b>	0.960" x 1.087" (2.438cm x 2.761cm)	0.960 x 1.297 (2.438cm x 3.294cm)

<b><i>Operating Temperature</i></b>	<b><i>-40 to 85° C (industrial)</i></b>	<b><i>-40 to 85° C (industrial)</i></b>
<b><i>Supported Network Topologies</i></b>	<b><i>Point-to-point, Point-to- multipoint, Peer-to- peer, and Mesh</i></b>	<b><i>Point-to-point, Point- to-multipoint, Peer-to- peer, and Mesh</i></b>
<b><i>Number of Channels</i></b>	<b><i>16 Direct Sequence Channels</i></b>	<b><i>14 Direct Sequence Channels</i></b>
<b><i>Channels</i></b>	<b><i>11 to 26</i></b>	<b><i>11 to 24</i></b>
<b><i>Addressing Options</i></b>	<b><i>PAN ID and Addresses, Cluster IDs</i></b>	<b><i>PAN ID and Addresses, Cluster IDs and Endpoints (optional)</i></b>

# \*IC 74HC595

Table 6. Static characteristics

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	-40 °C to +85 °C			-40 °C to +125 °C		Unit
			Min	Typ	Max	Min	Max	
74HC595								
V <sub>IH</sub>	HIGH-level input voltage	V <sub>CC</sub> = 2.0 V	1.5	1.2	-	1.5	-	V
		V <sub>CC</sub> = 4.5 V	3.15	2.4	-	3.15	-	V
		V <sub>CC</sub> = 6.0 V	4.2	3.2	-	4.2	-	V
V <sub>IL</sub>	LOW-level input voltage	V <sub>CC</sub> = 2.0 V	-	0.8	0.5	-	0.5	V
		V <sub>CC</sub> = 4.5 V	-	2.1	1.35	-	1.35	V
		V <sub>CC</sub> = 6.0 V	-	2.8	1.8	-	1.8	V
V <sub>OH</sub>	HIGH-level output voltage	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>						
		all outputs						
		I <sub>O</sub> = -20 µA; V <sub>CC</sub> = 2.0 V	1.9	2.0	-	1.9	-	V
		I <sub>O</sub> = -20 µA; V <sub>CC</sub> = 4.5 V	4.4	4.5	-	4.4	-	V
		I <sub>O</sub> = -20 µA; V <sub>CC</sub> = 6.0 V	5.9	6.0	-	5.9	-	V
		Q7S output						
		I <sub>O</sub> = -4 mA; V <sub>CC</sub> = 4.5 V	3.84	4.32	-	3.7	-	V
		I <sub>O</sub> = -5.2 mA; V <sub>CC</sub> = 6.0 V	5.34	5.81	-	5.2	-	V
		Qn bus driver outputs						
		I <sub>O</sub> = -6 mA; V <sub>CC</sub> = 4.5 V	3.84	4.32	-	3.7	-	V
		I <sub>O</sub> = -7.8 mA; V <sub>CC</sub> = 6.0 V	5.34	5.81	-	5.2	-	V
V <sub>OL</sub>	LOW-level output voltage	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>						
		all outputs						
		I <sub>O</sub> = 20 µA; V <sub>CC</sub> = 2.0 V	-	0	0.1	-	0.1	V
		I <sub>O</sub> = 20 µA; V <sub>CC</sub> = 4.5 V	-	0	0.1	-	0.1	V
		I <sub>O</sub> = 20 µA; V <sub>CC</sub> = 6.0 V	-	0	0.1	-	0.1	V
		Q7S output						
		I <sub>O</sub> = 4 mA; V <sub>CC</sub> = 4.5 V	-	0.15	0.33	-	0.4	V
		I <sub>O</sub> = 5.2 mA; V <sub>CC</sub> = 6.0 V	-	0.16	0.33	-	0.4	V
		Qn bus driver outputs						
		I <sub>O</sub> = 6 mA; V <sub>CC</sub> = 4.5 V	-	0.15	0.33	-	0.4	V
		I <sub>O</sub> = 7.8 mA; V <sub>CC</sub> = 6.0 V	-	0.16	0.33	-	0.4	V
I <sub>I</sub>	input leakage current	V <sub>I</sub> = V <sub>CC</sub> or GND; V <sub>CC</sub> = 6.0 V	-	-	±1.0	-	±1.0	µA
I <sub>oz</sub>	OFF-state output current	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub> ; V <sub>CC</sub> = 6.0 V; V <sub>O</sub> = V <sub>CC</sub> or GND	-	-	±5.0	-	±10	µA
I <sub>CC</sub>	supply current	V <sub>I</sub> = V <sub>CC</sub> or GND; I <sub>O</sub> = 0 A; V <sub>CC</sub> = 6.0 V	-	-	80	-	160	µA
C <sub>I</sub>	input capacitance		-	3.5	-	-	-	pF

Table 7. Dynamic characteristics

Voltages are referenced to GND (ground = 0 V); for test circuit see [Figure 13](#).

Symbol	Parameter	Conditions	25 °C			-40 °C to +85 °C		-40 °C to +125 °C		Unit
			Min	Typ <sup>[1]</sup>	Max	Min	Max	Min	Max	
74HC595										
t <sub>pd</sub>	propagation delay	SHCP to Q7S; see <a href="#">Figure 8</a> <sup>[2]</sup>								
		V <sub>CC</sub> = 2 V	-	52	160	-	200	-	240	ns
		V <sub>CC</sub> = 4.5 V	-	19	32	-	40	-	48	ns
		V <sub>CC</sub> = 6 V	-	15	27	-	34	-	41	ns
		STCP to Qn; see <a href="#">Figure 9</a> <sup>[2]</sup>								
		V <sub>CC</sub> = 2 V	-	55	175	-	220	-	265	ns
		V <sub>CC</sub> = 4.5 V	-	20	35	-	44	-	53	ns
		V <sub>CC</sub> = 6 V	-	16	30	-	37	-	45	ns
		MR to Q7S; see <a href="#">Figure 11</a> <sup>[2]</sup>								
		V <sub>CC</sub> = 2 V	-	47	175	-	220	-	265	ns
		V <sub>CC</sub> = 4.5 V	-	17	35	-	44	-	53	ns
		V <sub>CC</sub> = 6 V	-	14	30	-	37	-	45	ns
t <sub>en</sub>	enable time	OE to Qn; see <a href="#">Figure 12</a> <sup>[2]</sup>								
		V <sub>CC</sub> = 2 V	-	47	150	-	190	-	225	ns
		V <sub>CC</sub> = 4.5 V	-	17	30	-	38	-	45	ns
		V <sub>CC</sub> = 6 V	-	14	26	-	33	-	38	ns
t <sub>dis</sub>	disable time	OE to Qn; see <a href="#">Figure 12</a> <sup>[2]</sup>								
		V <sub>CC</sub> = 2 V	-	41	150	-	190	-	225	ns
		V <sub>CC</sub> = 4.5 V	-	15	30	-	38	-	45	ns
		V <sub>CC</sub> = 6 V	-	12	27	-	33	-	38	ns
t <sub>w</sub>	pulse width	SHCP HIGH or LOW; see <a href="#">Figure 8</a>								
		V <sub>CC</sub> = 2 V	75	17	-	95	-	110	-	ns
		V <sub>CC</sub> = 4.5 V	15	6	-	19	-	22	-	ns
		V <sub>CC</sub> = 6 V	13	5	-	16	-	19	-	ns
		STCP HIGH or LOW; see <a href="#">Figure 9</a>								
		V <sub>CC</sub> = 2 V	75	11	-	95	-	110	-	ns
		V <sub>CC</sub> = 4.5 V	15	4	-	19	-	22	-	ns
		V <sub>CC</sub> = 6 V	13	3	-	16	-	19	-	ns
		MR LOW; see <a href="#">Figure 11</a>								
		V <sub>CC</sub> = 2 V	75	17	-	95	-	110	-	ns
		V <sub>CC</sub> = 4.5 V	15	6	-	19	-	22	-	ns
		V <sub>CC</sub> = 6 V	13	5	-	16	-	19	-	ns

Table 7. Dynamic characteristics ...continued

Voltages are referenced to GND (ground = 0 V); for test circuit see Figure 13.

Symbol	Parameter	Conditions	25 °C			-40 °C to +85 °C		-40 °C to +125 °C		Unit
			Min	Typ <sup>(1)</sup>	Max	Min	Max	Min	Max	
t <sub>su</sub>	set-up time	DS to SHCP; see Figure 9								
		V <sub>CC</sub> = 2 V	50	11	-	65	-	75	-	ns
		V <sub>CC</sub> = 4.5 V	10	4	-	13	-	15	-	ns
		V <sub>CC</sub> = 6 V	9	3	-	11	-	13	-	ns
		SHCP to STCP; see Figure 10								
		V <sub>CC</sub> = 2 V	75	22	-	95	-	110	-	ns
		V <sub>CC</sub> = 4.5 V	15	8	-	19	-	22	-	ns
		V <sub>CC</sub> = 6 V	13	7	-	16	-	19	-	ns
t <sub>h</sub>	hold time	DS to SHCP; see Figure 10								
		V <sub>CC</sub> = 2 V	3	-6	-	3	-	3	-	ns
		V <sub>CC</sub> = 4.5 V	3	-2	-	3	-	3	-	ns
		V <sub>CC</sub> = 6 V	3	-2	-	3	-	3	-	ns
t <sub>rec</sub>	recovery time	MR to SHCP; see Figure 11								
		V <sub>CC</sub> = 2 V	50	-19	-	65	-	75	-	ns
		V <sub>CC</sub> = 4.5 V	10	-7	-	13	-	15	-	ns
		V <sub>CC</sub> = 6 V	9	-6	-	11	-	13	-	ns
f <sub>max</sub>	maximum frequency	SHCP or STCP; see Figure 8 and 9								
		V <sub>CC</sub> = 2 V	9	30	-	4.8	-	4	-	MHz
		V <sub>CC</sub> = 4.5 V	30	91	-	24	-	20	-	MHz
		V <sub>CC</sub> = 6 V	35	108	-	28	-	24	-	MHz
C <sub>PD</sub>	power dissipation capacitance	f <sub>i</sub> = 1 MHz; V <sub>i</sub> = GND to V <sub>CC</sub> <sup>(5)(7)</sup>	-	115	-	-	-	-	-	pF

## \*IC 74HC138

Table 6. Static characteristics

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	T <sub>amb</sub> = 25 °C			T <sub>amb</sub> = -40 °C to +85 °C		T <sub>amb</sub> = -40 °C to +125 °C		Unit
			Min	Typ	Max	Min	Max	Min	Max	
74HC138										
V <sub>IH</sub>	HIGH-level input voltage	V <sub>CC</sub> = 2.0 V	1.5	1.2	-	1.5	-	1.5	-	V
		V <sub>CC</sub> = 4.5 V	3.15	2.4	-	3.15	-	3.15	-	V
		V <sub>CC</sub> = 6.0 V	4.2	3.2	-	4.2	-	4.2	-	V
V <sub>IL</sub>	LOW-level input voltage	V <sub>CC</sub> = 2.0 V	-	0.8	0.5	-	0.5	-	0.5	V
		V <sub>CC</sub> = 4.5 V	-	2.1	1.35	-	1.35	-	1.35	V
		V <sub>CC</sub> = 6.0 V	-	2.8	1.8	-	1.8	-	1.8	V
V <sub>OH</sub>	HIGH-level output voltage	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>								
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 2.0 V	1.9	2.0	-	1.9	-	1.9	-	V
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 4.5 V	4.4	4.5	-	4.4	-	4.4	-	V
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 6.0 V	5.9	6.0	-	5.9	-	5.9	-	V
		I <sub>O</sub> = -4.0 mA; V <sub>CC</sub> = 4.5 V	3.98	4.32	-	3.84	-	3.7	-	V
		I <sub>O</sub> = -5.2 mA; V <sub>CC</sub> = 6.0 V	5.48	5.81	-	5.34	-	5.2	-	V
V <sub>OL</sub>	LOW-level output voltage	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>								
		I <sub>O</sub> = 20 μA; V <sub>CC</sub> = 2.0 V	-	0	0.1	-	0.1	-	0.1	V
		I <sub>O</sub> = 20 μA; V <sub>CC</sub> = 4.5 V	-	0	0.1	-	0.1	-	0.1	V
		I <sub>O</sub> = 20 μA; V <sub>CC</sub> = 6.0 V	-	0	0.1	-	0.1	-	0.1	V
		I <sub>O</sub> = 4.0 mA; V <sub>CC</sub> = 4.5 V	-	0.15	0.26	-	0.33	-	0.4	V
		I <sub>O</sub> = 5.2 mA; V <sub>CC</sub> = 6.0 V	-	0.16	0.26	-	0.33	-	0.4	V
I <sub>I</sub>	input leakage current	V <sub>I</sub> = V <sub>CC</sub> or GND; V <sub>CC</sub> = 6.0 V	-	-	±0.1	-	±1.0	-	±1.0	μA
I <sub>CC</sub>	supply current	V <sub>I</sub> = V <sub>CC</sub> or GND; I <sub>O</sub> = 0 A; V <sub>CC</sub> = 6.0 V	-	-	8.0	-	80	-	160	μA

Table 6. Static characteristics ...continued

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	T <sub>amb</sub> = 25 °C			T <sub>amb</sub> = -40 °C to +85 °C		T <sub>amb</sub> = -40 °C to +125 °C		Unit
			Min	Typ	Max	Min	Max	Min	Max	
C <sub>I</sub>	input capacitance		-	3.5	-					pF



Table 7. Dynamic characteristics

Voltages are referenced to GND (ground = 0 V);  $C_L = 50$  pF unless otherwise specified; for test circuit see Figure 8.

Symbol	Parameter	Conditions	T <sub>amb</sub> = 25 °C			T <sub>amb</sub> = -40 °C to +85 °C		T <sub>amb</sub> = -40 °C to +125 °C		Unit
			Min	Typ	Max	Min	Max	Min	Max	
74HC138										
t <sub>pd</sub>	propagation delay	An to $\overline{Y}_n$ ; see <a href="#">Figure 6</a> <a href="#">[1]</a>								
		V <sub>CC</sub> = 2.0 V	-	41	150	-	190	-	225	ns
		V <sub>CC</sub> = 4.5 V	-	15	30	-	38	-	45	ns
		V <sub>CC</sub> = 5 V; C <sub>L</sub> = 15 pF	-	12	-	-	-	-	-	ns
		V <sub>CC</sub> = 6.0 V	-	12	26	-	33	-	38	ns
		E3 to $\overline{Y}_n$ ; see <a href="#">Figure 6</a> <a href="#">[1]</a>								
		V <sub>CC</sub> = 2.0 V	-	47	150	-	190	-	225	ns
		V <sub>CC</sub> = 4.5 V	-	17	20	-	38	-	45	ns
		V <sub>CC</sub> = 5 V; C <sub>L</sub> = 15 pF	-	14	-	-	-	-	-	ns
		V <sub>CC</sub> = 6.0 V	-	14	26	-	33	-	38	ns
		$\overline{E}_n$ to $\overline{Y}_n$ ; see <a href="#">Figure 7</a> <a href="#">[1]</a>								
		V <sub>CC</sub> = 2.0 V	-	47	150	-	190	-	225	ns
		V <sub>CC</sub> = 4.5 V	-	17	20	-	38	-	45	ns
		V <sub>CC</sub> = 5 V; C <sub>L</sub> = 15 pF	-	14	-	-	-	-	-	ns
		V <sub>CC</sub> = 6.0 V	-	14	26	-	33	-	38	ns
t <sub>t</sub>	transition time	$\overline{Y}_n$ ; see <a href="#">Figure 6</a> and <a href="#">Figure 7</a> <a href="#">[2]</a>								
		V <sub>CC</sub> = 2.0 V	-	19	75	-	95	-	110	ns
		V <sub>CC</sub> = 4.5 V	-	7	15	-	19	-	22	ns
		V <sub>CC</sub> = 6.0 V	-	6	13	-	16	-	19	ns
C <sub>PD</sub>	power dissipation capacitance	C <sub>L</sub> = 50 pF; f = 1 MHz; V <sub>I</sub> = GND to V <sub>CC</sub> <a href="#">[3]</a>	-	67	-	-	-	-	pF	

## \*IC 74HC245

Table 7: Static characteristics type 74HC245

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
T <sub>amb</sub> = 25 °C						
V <sub>IH</sub>	HIGH-level input voltage	V <sub>CC</sub> = 2.0 V	1.5	1.2	-	V
		V <sub>CC</sub> = 4.5 V	3.15	2.4	-	V
		V <sub>CC</sub> = 6.0 V	4.2	3.2	-	V
V <sub>IL</sub>	LOW-level input voltage	V <sub>CC</sub> = 2.0 V	-	0.8	0.5	V
		V <sub>CC</sub> = 4.5 V	-	2.1	1.35	V
		V <sub>CC</sub> = 6.0 V	-	2.8	1.8	V
V <sub>OH</sub>	HIGH-level output voltage	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>				
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 2.0 V	1.9	2.0	-	V
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 4.5 V	4.4	4.5	-	V
		I <sub>O</sub> = -20 μA; V <sub>CC</sub> = 6.0 V	5.9	6.0	-	V
		I <sub>O</sub> = -6.0 mA; V <sub>CC</sub> = 4.5 V	3.98	4.32	-	V
		I <sub>O</sub> = -7.8 mA; V <sub>CC</sub> = 6.0 V	5.48	5.81	-	V

Table 7: Static characteristics type 74HC245 ...continued

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{OL}$	LOW-level output voltage	$V_I = V_{IH}$ or $V_{IL}$				
		$I_O = 20\ \mu\text{A}; V_{CC} = 2.0\ \text{V}$	-	0	0.1	V
		$I_O = 20\ \mu\text{A}; V_{CC} = 4.5\ \text{V}$	-	0	0.1	V
		$I_O = 20\ \mu\text{A}; V_{CC} = 6.0\ \text{V}$	-	0	0.1	V
		$I_O = 6.0\ \text{mA}; V_{CC} = 4.5\ \text{V}$	-	0.15	0.26	V
		$I_O = 7.8\ \text{mA}; V_{CC} = 6.0\ \text{V}$	-	0.16	0.26	V
$I_{LI}$	input leakage current	$V_I = V_{CC}$ or GND; $V_{CC} = 6.0\ \text{V}$	-	-	$\pm 0.1$	$\mu\text{A}$
$I_{OZ}$	OFF-state output current	$V_I = V_{IH}$ or $V_{IL}$ ; $V_O = V_{CC}$ or GND; $V_{CC} = 6.0\ \text{V}$	-	-	$\pm 0.5$	$\mu\text{A}$
$I_{CC}$	quiescent supply current	$V_I = V_{CC}$ or GND; $I_O = 0\ \text{A}$ ; $V_{CC} = 6.0\ \text{V}$	-	-	8.0	$\mu\text{A}$
$C_I$	input capacitance		-	3.5	-	pF
$C_{IO}$	input/output capacitance		-	10	-	pF
$T_{\text{amb}} = -40\ ^\circ\text{C}$ to $+85\ ^\circ\text{C}$						
$V_{IH}$	HIGH-level input voltage	$V_{CC} = 2.0\ \text{V}$	1.5	-	-	V
		$V_{CC} = 4.5\ \text{V}$	3.15	-	-	V
		$V_{CC} = 6.0\ \text{V}$	4.2	-	-	V
$V_{IL}$	LOW-level input voltage	$V_{CC} = 2.0\ \text{V}$	-	-	0.5	V
		$V_{CC} = 4.5\ \text{V}$	-	-	1.35	V
		$V_{CC} = 6.0\ \text{V}$	-	-	1.8	V
$V_{OH}$	HIGH-level output voltage	$V_I = V_{IH}$ or $V_{IL}$				
		$I_O = -20\ \mu\text{A}; V_{CC} = 2.0\ \text{V}$	1.9	-	-	V
		$I_O = -20\ \mu\text{A}; V_{CC} = 4.5\ \text{V}$	4.4	-	-	V
		$I_O = -20\ \mu\text{A}; V_{CC} = 6.0\ \text{V}$	5.9	-	-	V
		$I_O = -6.0\ \text{mA}; V_{CC} = 4.5\ \text{V}$	3.84	-	-	V
		$I_O = -7.8\ \text{mA}; V_{CC} = 6.0\ \text{V}$	5.34	-	-	V
$V_{OL}$	LOW-level output voltage	$V_I = V_{IH}$ or $V_{IL}$				
		$I_O = 20\ \mu\text{A}; V_{CC} = 2.0\ \text{V}$	-	-	0.1	V
		$I_O = 20\ \mu\text{A}; V_{CC} = 4.5\ \text{V}$	-	-	0.1	V
		$I_O = 20\ \mu\text{A}; V_{CC} = 6.0\ \text{V}$	-	-	0.1	V
		$I_O = 6.0\ \text{mA}; V_{CC} = 4.5\ \text{V}$	-	-	0.33	V
		$I_O = 7.8\ \text{mA}; V_{CC} = 6.0\ \text{V}$	-	-	0.33	V
$I_{LI}$	input leakage current	$V_I = V_{CC}$ or GND; $V_{CC} = 6.0\ \text{V}$	-	-	$\pm 1.0$	$\mu\text{A}$
$I_{OZ}$	OFF-state output current	$V_I = V_{IH}$ or $V_{IL}$ ; $V_O = V_{CC}$ or GND; $V_{CC} = 6.0\ \text{V}$	-	-	$\pm 5.0$	$\mu\text{A}$
$I_{CC}$	quiescent supply current	$V_I = V_{CC}$ or GND; $I_O = 0\ \text{A}$ ; $V_{CC} = 6.0\ \text{V}$	-	-	80	$\mu\text{A}$
$T_{\text{amb}} = -40\ ^\circ\text{C}$ to $+125\ ^\circ\text{C}$						
$V_{IH}$	HIGH-level input voltage	$V_{CC} = 2.0\ \text{V}$	1.5	-	-	V
		$V_{CC} = 4.5\ \text{V}$	3.15	-	-	V
		$V_{CC} = 6.0\ \text{V}$	4.2	-	-	V

Table 7: Static characteristics type 74HC245 ...continued

At recommended operating conditions; voltages are referenced to GND (ground = 0 V).

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{IL}$	LOW-level input voltage	$V_{CC} = 2.0 \text{ V}$	-	-	0.5	V
		$V_{CC} = 4.5 \text{ V}$	-	-	1.35	V
		$V_{CC} = 6.0 \text{ V}$	-	-	1.8	V
$V_{OH}$	HIGH-level output voltage	$V_I = V_{IH}$ or $V_{IL}$	-	-	-	
		$I_O = -20 \mu\text{A}; V_{CC} = 2.0 \text{ V}$	1.9	-	-	V
		$I_O = -20 \mu\text{A}; V_{CC} = 4.5 \text{ V}$	4.4	-	-	V
		$I_O = -20 \mu\text{A}; V_{CC} = 6.0 \text{ V}$	5.9	-	-	V
		$I_O = -8.0 \text{ mA}; V_{CC} = 4.5 \text{ V}$	3.7	-	-	V
		$I_O = -7.8 \text{ mA}; V_{CC} = 6.0 \text{ V}$	5.2	-	-	V
$V_{OL}$	LOW-level output voltage	$V_I = V_{IH}$ or $V_{IL}$	-	-	-	
		$I_O = 20 \mu\text{A}; V_{CC} = 2.0 \text{ V}$	-	-	0.1	V
		$I_O = 20 \mu\text{A}; V_{CC} = 4.5 \text{ V}$	-	-	0.1	V
		$I_O = 20 \mu\text{A}; V_{CC} = 6.0 \text{ V}$	-	-	0.1	V
		$I_O = 6.0 \text{ mA}; V_{CC} = 4.5 \text{ V}$	-	-	0.4	V
		$I_O = 7.8 \text{ mA}; V_{CC} = 6.0 \text{ V}$	-	-	0.4	V
$I_{II}$	input leakage current	$V_I = V_{CC}$ or GND; $V_{CC} = 6.0 \text{ V}$	-	-	$\pm 1.0$	$\mu\text{A}$
$I_{OZ}$	OFF-state output current	$V_I = V_{IH}$ or $V_{IL}; V_O = V_{CC}$ or GND; $V_{CC} = 6.0 \text{ V}$	-	-	$\pm 10.0$	$\mu\text{A}$
$I_{CC}$	quiescent supply current	$V_I = V_{CC}$ or GND; $I_O = 0 \text{ A};$ $V_{CC} = 6.0 \text{ V}$	-	-	160	$\mu\text{A}$

Table 9: Dynamic characteristics type 74HC245

GND = 0 V; test circuit see Figure 7.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$T_{amb} = 25 \text{ }^\circ\text{C}$						
$t_{PHL}, t_{PLH}$	propagation delay An to Bn or Bn to An	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	25	90	ns
		$V_{CC} = 4.5 \text{ V}$	-	9	18	ns
		$V_{CC} = 6.0 \text{ V}; C_L = 15 \text{ pF}$	-	7	-	ns
		$V_{CC} = 6.0 \text{ V}$	-	7	15	ns
$t_{PZH}, t_{PZL}$	3-state output enable time $\overline{OE}$ to An or OE to Bn	see Figure 6				
		$V_{CC} = 2.0 \text{ V}$	-	30	150	ns
		$V_{CC} = 4.5 \text{ V}$	-	11	30	ns
		$V_{CC} = 6.0 \text{ V}$	-	9	26	ns
$t_{PHZ}, t_{PLZ}$	3-state output disable time $\overline{OE}$ to An or OE to Bn	see Figure 6				
		$V_{CC} = 2.0 \text{ V}$	-	41	150	ns
		$V_{CC} = 4.5 \text{ V}$	-	15	30	ns
		$V_{CC} = 6.0 \text{ V}$	-	12	26	ns
$t_{THL}, t_{TLH}$	output transition time	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	14	60	ns
		$V_{CC} = 4.5 \text{ V}$	-	5	12	ns
		$V_{CC} = 6.0 \text{ V}$	-	4	10	ns
$C_{PD}$	power dissipation capacitance per transceiver	$V_I = \text{GND to } V_{CC}$	11	30	-	pF
$T_{amb} = -40 \text{ }^\circ\text{C to } +85 \text{ }^\circ\text{C}$						
$t_{PHL}, t_{PLH}$	propagation delay An to Bn or Bn to An	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	-	115	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	23	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	20	ns

Table 9: Dynamic characteristics type 74HC245 ...continued  
GND = 0 V; test circuit see Figure 7.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$t_{PZH}, t_{PZL}$	3-state output enable time $\overline{OE}$ to An or $\overline{OE}$ to Bn	see Figure 8				
		$V_{CC} = 2.0 \text{ V}$	-	-	190	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	38	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	33	ns
$t_{PHZ}, t_{PLZ}$	3-state output disable time $\overline{OE}$ to An or $\overline{OE}$ to Bn	see Figure 8				
		$V_{CC} = 2.0 \text{ V}$	-	-	190	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	38	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	33	ns
$t_{rHL}, t_{rLH}$	output transition time	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	-	75	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	15	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	13	ns
$T_{amb} = -40\text{ }^{\circ}\text{C to } +125\text{ }^{\circ}\text{C}$						
$t_{PHL}, t_{PLH}$	propagation delay An to Bn or Bn to An	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	-	135	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	27	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	23	ns
$t_{PZH}, t_{PZL}$	3-state output enable time $\overline{OE}$ to An or $\overline{OE}$ to Bn	see Figure 8				
		$V_{CC} = 2.0 \text{ V}$	-	-	225	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	45	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	38	ns
$t_{PHZ}, t_{PLZ}$	3-state output disable time $\overline{OE}$ to An or $\overline{OE}$ to Bn	see Figure 8				
		$V_{CC} = 2.0 \text{ V}$	-	-	225	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	45	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	38	ns
$t_{rHL}, t_{rLH}$	output transition time	see Figure 5				
		$V_{CC} = 2.0 \text{ V}$	-	-	90	ns
		$V_{CC} = 4.5 \text{ V}$	-	-	18	ns
		$V_{CC} = 6.0 \text{ V}$	-	-	15	ns

[1]  $C_{PD}$  is used to determine the dynamic power dissipation ( $P_D$  in  $\mu\text{W}$ ):

$$P_D = C_{PD} \times V_{CC}^2 \times f_i \times N + \sum (C_L \times V_{CC}^2 \times f_o) \text{ where:}$$

$f_i$  = input frequency in MHz;

$f_o$  = output frequency in MHz;

$C_L$  = output load capacitance in pF;

$V_{CC}$  = supply voltage in V;

$N$  = number of inputs switching;

$\sum (C_L \times V_{CC}^2 \times f_o)$  = sum of outputs.

## \*IC APM4953

### Electrical Characteristics (T<sub>A</sub>=25°C unless otherwise noted)

Symbol	Parameter	Test Condition	APM4953			Unit
			Min.	Typ <sup>a</sup>	Max.	
Static						
BV <sub>DSS</sub>	Drain-Source Breakdown Voltage	V <sub>GS</sub> =0V, I <sub>DS</sub> =-250μA	-30			V
I <sub>DSS</sub>	Zero Gate Voltage Drain Current	V <sub>DS</sub> =-24V, V <sub>GS</sub> =0V			-1	μA
V <sub>GS(th)</sub>	Gate Threshold Voltage	V <sub>DS</sub> =V <sub>GS</sub> , I <sub>DS</sub> =-250μA	-1	-1.5	-2	V
I <sub>GSS</sub>	Gate Leakage Current	V <sub>GS</sub> =±25V, V <sub>DS</sub> =0V			±100	nA
R <sub>DS(ON)</sub>	Drain-Source On-state Resistance <sup>b</sup>	V <sub>GS</sub> =-10V, I <sub>DS</sub> =-4.9A		53	60	mΩ
		V <sub>GS</sub> =-4.5V, I <sub>DS</sub> =-3.6A		80	95	
V <sub>SD</sub>	Diode Forward Voltage <sup>b</sup>	I <sub>SD</sub> =-1.7A, V <sub>GS</sub> =0V		-0.7	-1.3	V
Dynamic <sup>a</sup>						
Q <sub>g</sub>	Total Gate Charge	V <sub>DS</sub> =-15V, I <sub>GS</sub> =-10V		22.3	29	nC
Q <sub>gs</sub>	Gate-Source Charge	I <sub>g</sub> =-4.6A		4.65		
Q <sub>gd</sub>	Gate-Drain Charge			2		
t <sub>d(ON)</sub>	Turn-on Delay Time	V <sub>DD</sub> =-15V, I <sub>g</sub> =-2A,		10	18	ns
T <sub>r</sub>	Turn-on Rise Time	V <sub>GEN</sub> =-10V, R <sub>G</sub> =6Ω		15	20	
t <sub>d(OFF)</sub>	Turn-off Delay Time	R <sub>L</sub> =7.5Ω		22	38	
T <sub>f</sub>	Turn-off Fall Time			15	25	
C <sub>iss</sub>	Input Capacitance	V <sub>GS</sub> =0V		1260		pF
C <sub>OSS</sub>	Output Capacitance	V <sub>DS</sub> =-25V		340		
C <sub>rss</sub>	Reverse Transfer Capacitance	Frequency=1.0MHz		220		

#### Notes

<sup>a</sup>: Pulse test; pulse width ≤300μs, duty cycle ≤ 2%

<sup>b</sup>: Guaranteed by design, not subject to production testing

## \*IC SN74HC04

### 6.4 Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS	V <sub>CC</sub>	T <sub>A</sub> = 25°C			SN54HC04		SN74HC04		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
V <sub>OH</sub>	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>	I <sub>OH</sub> = -20 μA	2 V	1.9	1.998	1.9		1.9		V
			4.5 V	4.4	4.499	4.4		4.4		
			6 V	5.9	5.999	5.9		5.9		
		I <sub>OH</sub> = -4 mA	4.5 V	3.98	4.3	3.7		3.84		
		I <sub>OH</sub> = -5.2 mA	6 V	5.48	5.8	5.2		5.34		
V <sub>OL</sub>	V <sub>I</sub> = V <sub>IH</sub> or V <sub>IL</sub>	I <sub>OL</sub> = 20 μA	2 V		0.002 0.1		0.1		0.1	V
			4.5 V		0.001 0.1		0.1		0.1	
			6 V		0.001 0.1		0.1		0.1	
		I <sub>OL</sub> = 4 mA	4.5 V		0.17 0.26		0.4		0.33	
		I <sub>OL</sub> = 5.2 mA	6 V		0.15 0.26		0.4		0.33	
I <sub>I</sub>	V <sub>I</sub> = V <sub>CC</sub> or 0	6 V		±0.1	±100		±1000		±1000	nA
I <sub>CC</sub>	V <sub>I</sub> = V <sub>CC</sub> or 0, I <sub>O</sub> = 0	6 V			2		40		20	μA
C <sub>i</sub>		6 V			3 10		10		10	pF

## \*XBee S2C

### Performance specifications

This table describes the performance specifications for the devices.

Specification	XBee ZigBee S2C	XBee-PRO ZigBee S2C	XBee ZigBee S2D
Indoor/urban range	Up to 60 m (200 ft)	Up to 90 m (300 ft)	Up to 60 m (200 ft)
Outdoor RF line-of-sight range	Up to 1200 m (4000 ft)	Up to 3200 m (2 mi)	Up to 1200 m (4000 ft)
Transmit power output (maximum)	6.3 mW (+8 dBm), boost mode 3.1 mW (+5 dBm), normal mode channel 26 max power is +3 dBm	63 mW (+18 dBm)	6.3 mW (+8 dBm) channel 26 max power is +1 dBm
RF data rate	250,000 b/s		
Receiver sensitivity	-102 dBm, boost mode -100 dBm, normal mode	-101 dBm	-102 dBm, boost mode -100 dBm, normal mode

### Power requirements

The following table describes the power requirements for the devices.

Specification	XBee ZigBee S2C	XBee-PRO ZigBee S2C	XBee ZigBee S2D
Adjustable power	Yes		
Supply voltage	2.1 - 3.6 V 2.2 - 3.6 V for programmable version	2.7 - 3.6 V	2.1 - 3.6 V
Operating current (transmit)	45 mA (+8 dBm, boost mode) 33 mA (+5 dBm, normal mode)	120 mA @ +3.3 V, +18 dBm	45 mA
Operating current (receive)	31 mA (boost mode) 28 mA (normal mode)	31 mA	31 mA
Power-down current	< 1 $\mu$ A @ 25°C		< 3 $\mu$ A @ 25°C

### General specifications

The following table describes the general specifications for the devices.

Specification	XBee ZigBee S2C	XBee-PRO ZigBee S2C	XBee ZigBee S2D
Operating frequency band	ISM 2.4 - 2.5 GHz		
Form factor	through-hole, surface-mount		surface-mount

Specification	XBee ZigBee S2C	XBee-PRO ZigBee S2C	XBee ZigBee S2D
Dimensions	through-hole: 2.438 x 2.761 cm (0.960 x 1.087 in) surface-mount: 2.199 x 3.4 x 0.305 cm (0.866 x 1.33 x 0.120 in)	through-hole: 2.438 x 3.294 cm (0.960 x 1.297 in) surface-mount: 2.199 x 3.4 x 0.305 cm (0.866 x 1.33 x 0.120 in)	surface-mount: 2.199 x 3.4 x 0.305 cm (0.866 x 1.33 x 0.120 in)
Operating temperature	-40 to 85 °C (industrial)		
Antenna options	through-hole: PCB antenna, U.FL connector, RPSMA connector, or integrated wire surface-mount: RF pad, PCB antenna, or U.FL connector		

## Networking and security specifications

The following table describes the networking and security specifications for the devices.

Specification	XBee ZigBee S2C	XBee-PRO ZigBee S2C	XBee ZigBee S2D
Supported network topologies	Point-to-point, point-to-multipoint, peer-to-peer, and DigiMesh		
Number of channels	16 Direct sequence channels	15 Direct sequence channels	16 Direct sequence channels
Interface immunity	Direct Sequence Spread Spectrum (DSSS)		
Channels	11 to 26		
Addressing options	PAN ID and addresses, cluster IDs and endpoints (optional)		

-----Halaman ini sengaja dikosongkan-----



## RIWAYAT HIDUP PENULIS



Nama : Moch Nafi Rizky K  
TTL : Surabaya, 03 November 1995  
Jenis Kelamin : Laki-Laki  
Agama : Islam  
Alamat Rumah : Jl. Medokan Semampir Indah 117 Surabaya  
Telp/HP : 085648142383  
E-mail : Nafirizky1@gmail.com  
Hobi : Membaca, menulis dan *traveling*

## RIWAYAT PENDIDIKAN

- 2002-2008 : SD Negeri 1 261
- 2008-2011 : SMP Negeri 19 Surabaya
- 2011-2014 : SMA Negeri 2 Surabaya
- 2014 – sekarang : Bidang Studi Komputer Kontrol, Departemen Teknik Elektro Otomasi, ITS

## PENGALAMAN KERJA

- Kerja Praktek di PT CNC Controller Indonesia (2016)

## PENGALAMAN ORGANISASI

- Lembaga Dakwah Jurusan Mahasiswa Departemen Teknik Elektro Otomasi ITS (2016-2017)

## PENGALAMAN KEPANITIAAN

- Panitia Keterampilan Manajemen Mahasiswa Pra-Tingkat Dasar BEM FTI- ITS (2014)
- Panitia Dana Industrial Automation and Robotic Competition (2015-2016 )
- Panitia Pelatihan Jurnalistik Tingkat Dasar Departemen Teknik Elektro Otomasi ITS (2016)

-----Halaman ini sengaja dikosongkan-----